(12) UK Patent Application (19) GB (11) 2 193 827 (13) A

(43) Application published 17 Feb 1988

(21) Application No 8717453

(22) Date of filing 23 Jul 1987

(30) Priority data
(31) 889513 (32) 25 Jul 1986 (33) US
29772 24 Mar 1987

(51) INT CL⁴
G06K 9/22 G06F 15/21 15/42 G06K 9/62

(52) Domestic classification (Edition J):
G4R 10E 10X 11A 11C 11D 11E 11F 12A 12F 1C 1D 1X 3B 3C 3E 3G 5B 5X 8A 8G 9B 9C EV HA RV
G4H 13D 14A 14B 14D 1A TAT
U1S 2104 2126 G4H G4R

(56) Documents cited
GB A 2102944 EP A2 0157354

(58) Field of search
G4R
Selected US specifications from IPC sub-class G06K

(71) Applicant
Linus Technologies Incorporated

(Incorporated in USA–Delaware)

1889 Preston White Drive, Reston, Virginia 22091, United States of America

(72) Inventor
Ralph Sklarew

(74) Agent and/or Address for Service
Reddie & Grose,
16 Theobalds Road, London WC1X 8PL

(54) **Handwritten keyboardless-entry computer system**

(57) A keyboardless-entry computer system includes a transparent input screen (18) that generates positional information when contacted by a stylus (16), and a display screen (20) mounted below the input screen (18) such that a character that is displayed can be seen below the input screen (18). The system includes a computer (14) programmed to compile the positional information into Strokes, to calculate Stroke characteristics, and then compare the Stroke characteristics with those stored in a database in order to recognise the symbol drawn by the stylus (16). The system may be used to edit text or complete standard forms, e.g. with medical data.

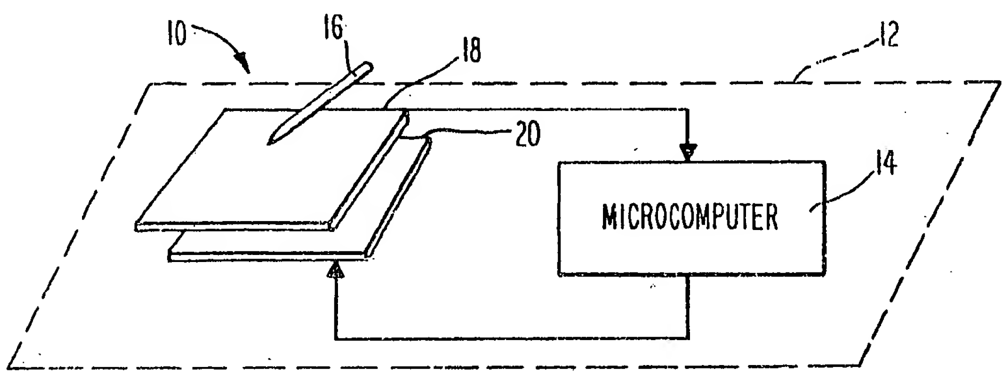

Fig. 1

GB 2 193 827 A

The drawing(s) originally filed was/were informal and the print here reproduced is taken from a later filed formal copy.

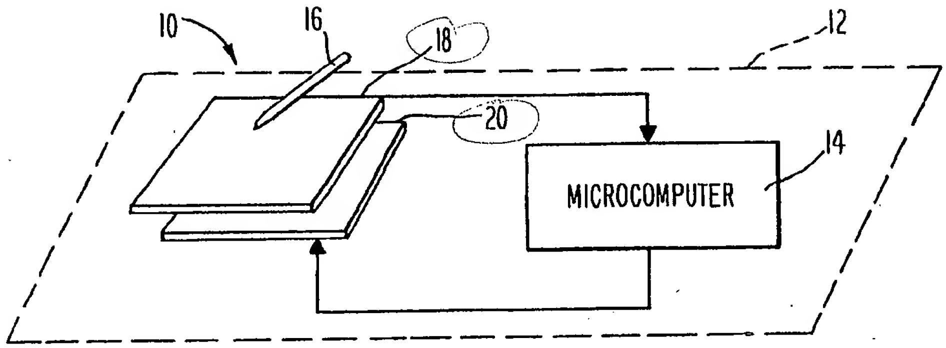

Fig. 1

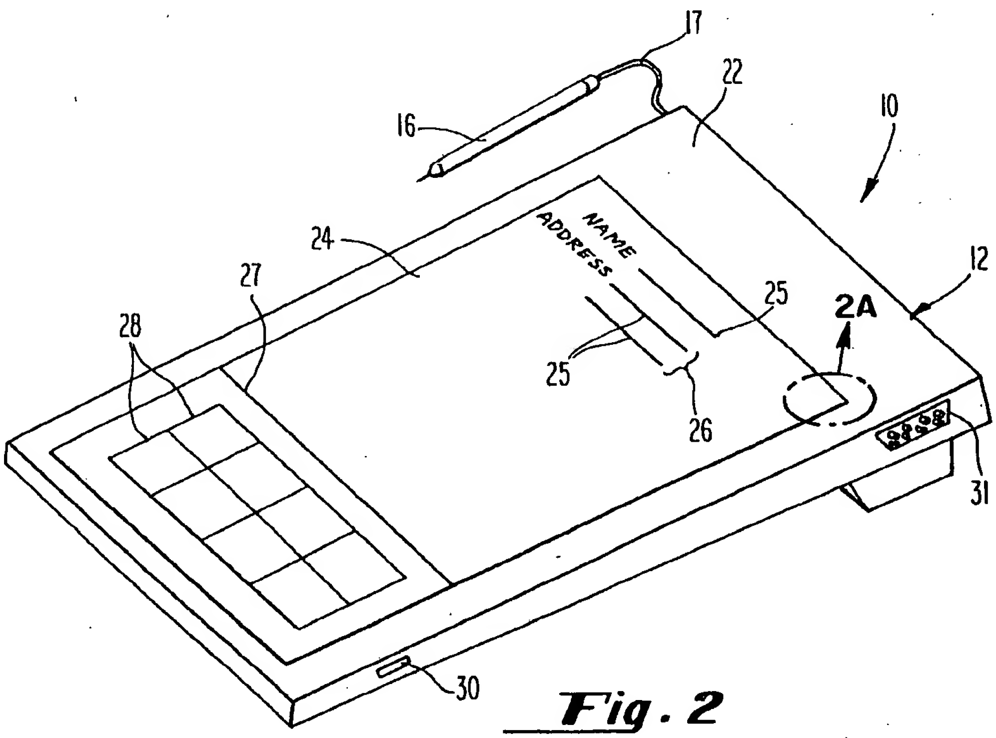

Fig. 2

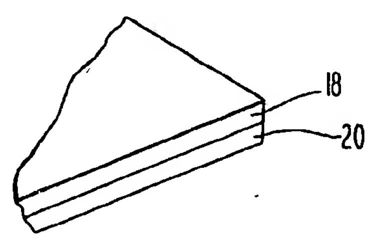

Fig. 2A

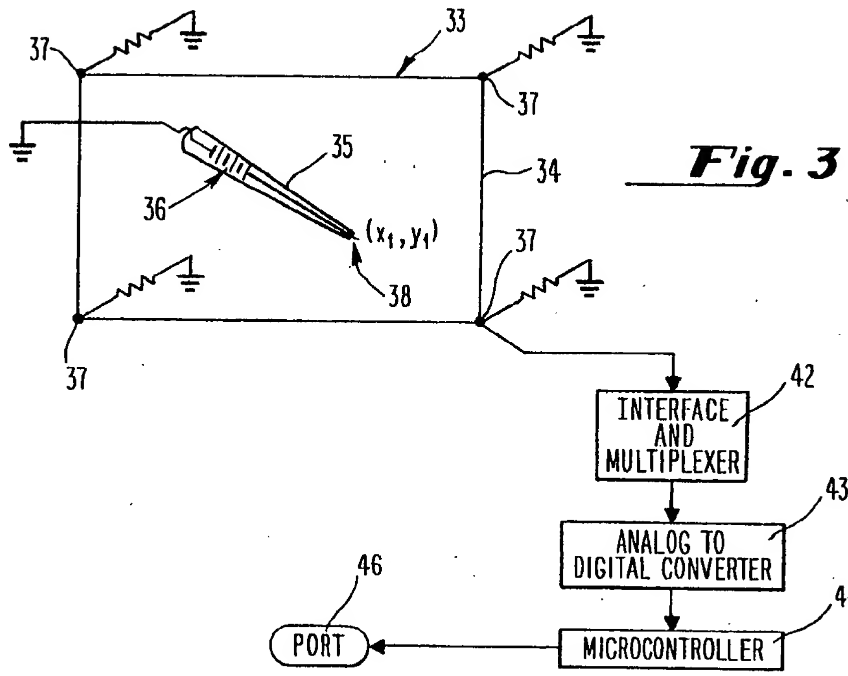
33
37
37
37
37
35
36
34
(x_1, y_1)
38
INTERFACE AND MULTIPLEXER
42
ANALOG TO DIGITAL CONVERTER
43
MICROCONTROLLER
44
PORT
46

Fig. 3

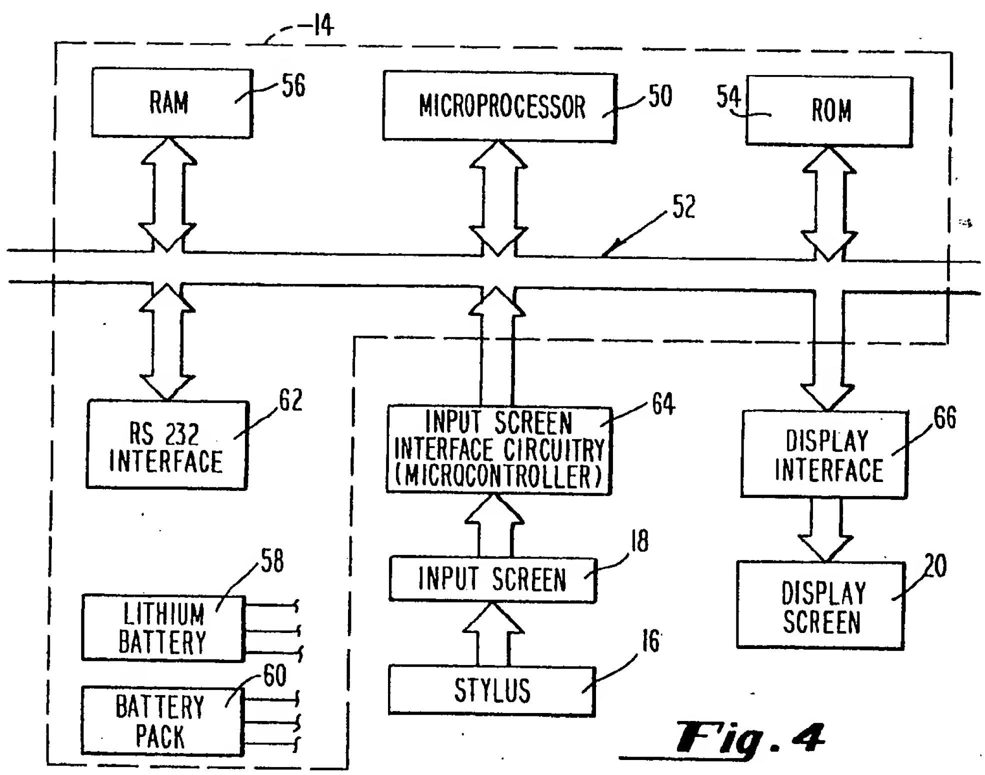
14
RAM
56
MICROPROCESSOR
50
ROM
54
52
RS 232 INTERFACE
62
INPUT SCREEN INTERFACE CIRCUITRY (MICROCONTROLLER)
64
DISPLAY INTERFACE
66
INPUT SCREEN
18
DISPLAY SCREEN
20
LITHIUM BATTERY
58
BATTERY PACK
60
STYLUS
16

Fig. 4

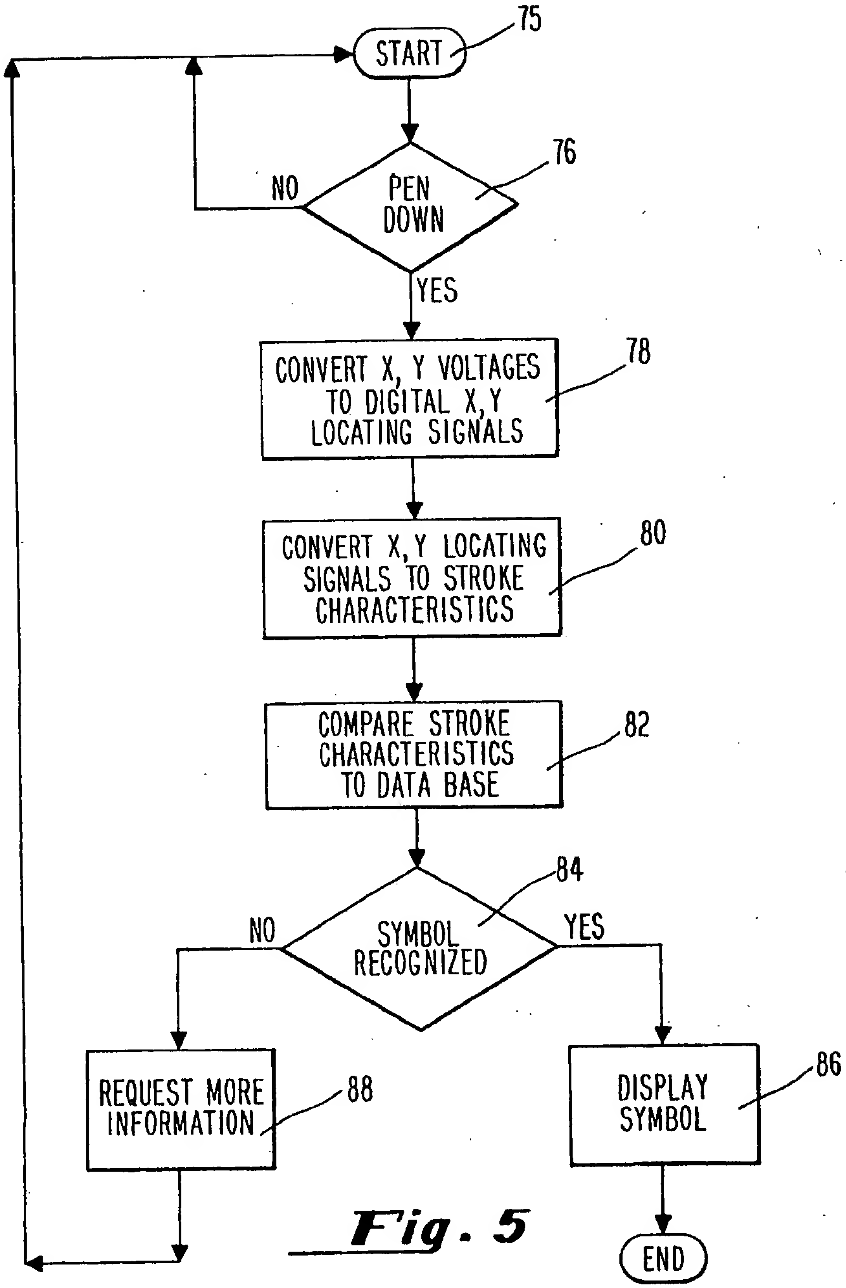
START
75
PEN DOWN
76
NO
YES
CONVERT X, Y VOLTAGES TO DIGITAL X,Y LOCATING SIGNALS
78
CONVERT X, Y LOCATING SIGNALS TO STROKE CHARACTERISTICS
80
COMPARE STROKE CHARACTERISTICS TO DATA BASE
82
SYMBOL RECOGNIZED
84
NO
YES
REQUEST MORE INFORMATION
88
DISPLAY SYMBOL
86
END

Fig. 5

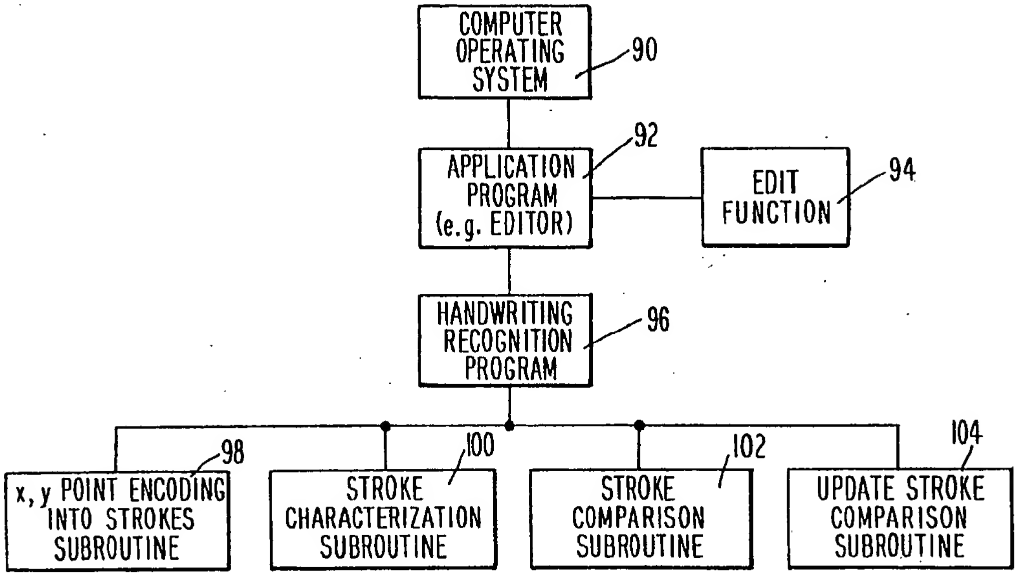
COMPUTER OPERATING SYSTEM
90
APPLICATION PROGRAM (e.g. EDITOR)
92
EDIT FUNCTION
94
HANDWRITING RECOGNITION PROGRAM
96
x, y POINT ENCODING INTO STROKES SUBROUTINE
98
STROKE CHARACTERIZATION SUBROUTINE
100
STROKE COMPARISON SUBROUTINE
102
UPDATE STROKE COMPARISON SUBROUTINE
104

Fig. 6

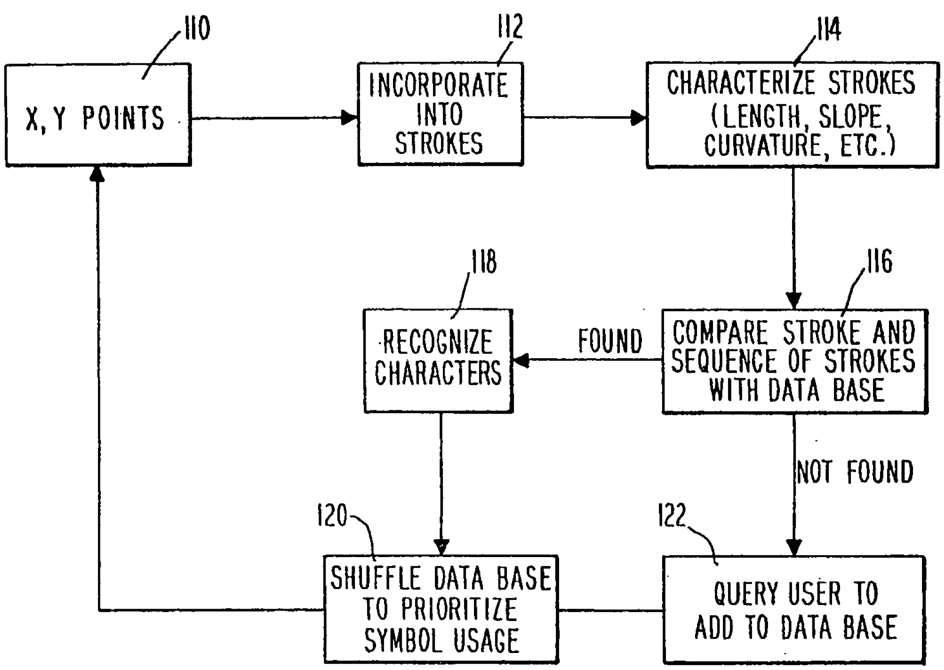
X, Y POINTS
110
INCORPORATE INTO STROKES
112
CHARACTERIZE STROKES (LENGTH, SLOPE, CURVATURE, ETC.)
114
RECOGNIZE CHARACTERS
118
FOUND
COMPARE STROKE AND SEQUENCE OF STROKES WITH DATA BASE
116
NOT FOUND
120
SHUFFLE DATA BASE TO PRIORITIZE SYMBOL USAGE
122
QUERY USER TO ADD TO DATA BASE

Fig. 7

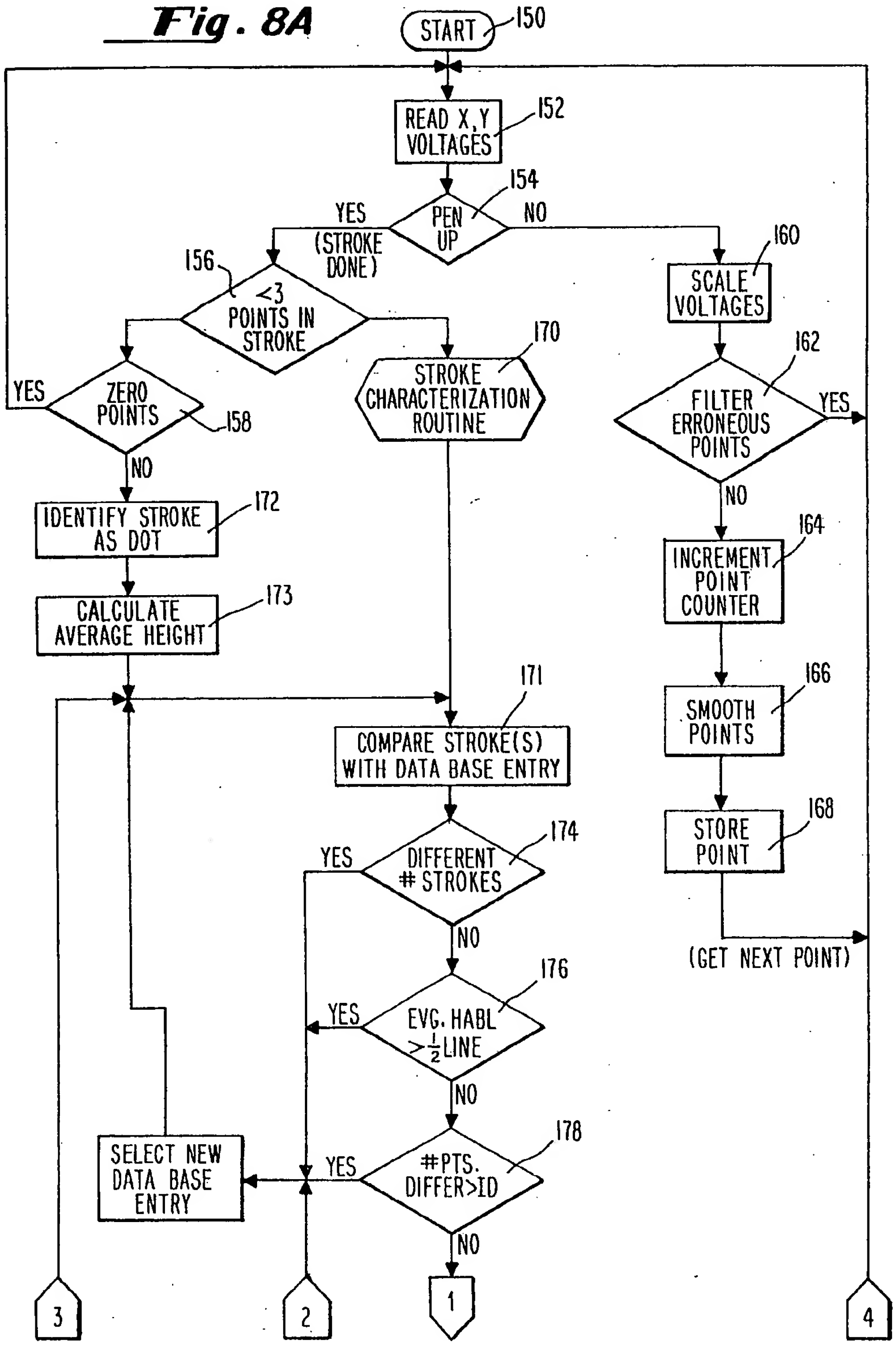
START
150
READ X,Y VOLTAGES
152
PEN UP
154
YES (STROKE DONE)
NO
156
<3 POINTS IN STROKE
ZERO POINTS
158
YES
NO
IDENTIFY STROKE AS DOT
172
CALCULATE AVERAGE HEIGHT
173
STROKE CHARACTERIZATION ROUTINE
170
SCALE VOLTAGES
160
FILTER ERRONEOUS POINTS
162
YES
NO
INCREMENT POINT COUNTER
164
SMOOTH POINTS
166
STORE POINT
168
(GET NEXT POINT)
COMPARE STROKE(S) WITH DATA BASE ENTRY
171
DIFFERENT # STROKES
174
YES
NO
EVG. HABL >½LINE
176
YES
NO
#PTS. DIFFER>ID
178
YES
NO
SELECT NEW DATA BASE ENTRY
3
2
1
4

Fig. 8A

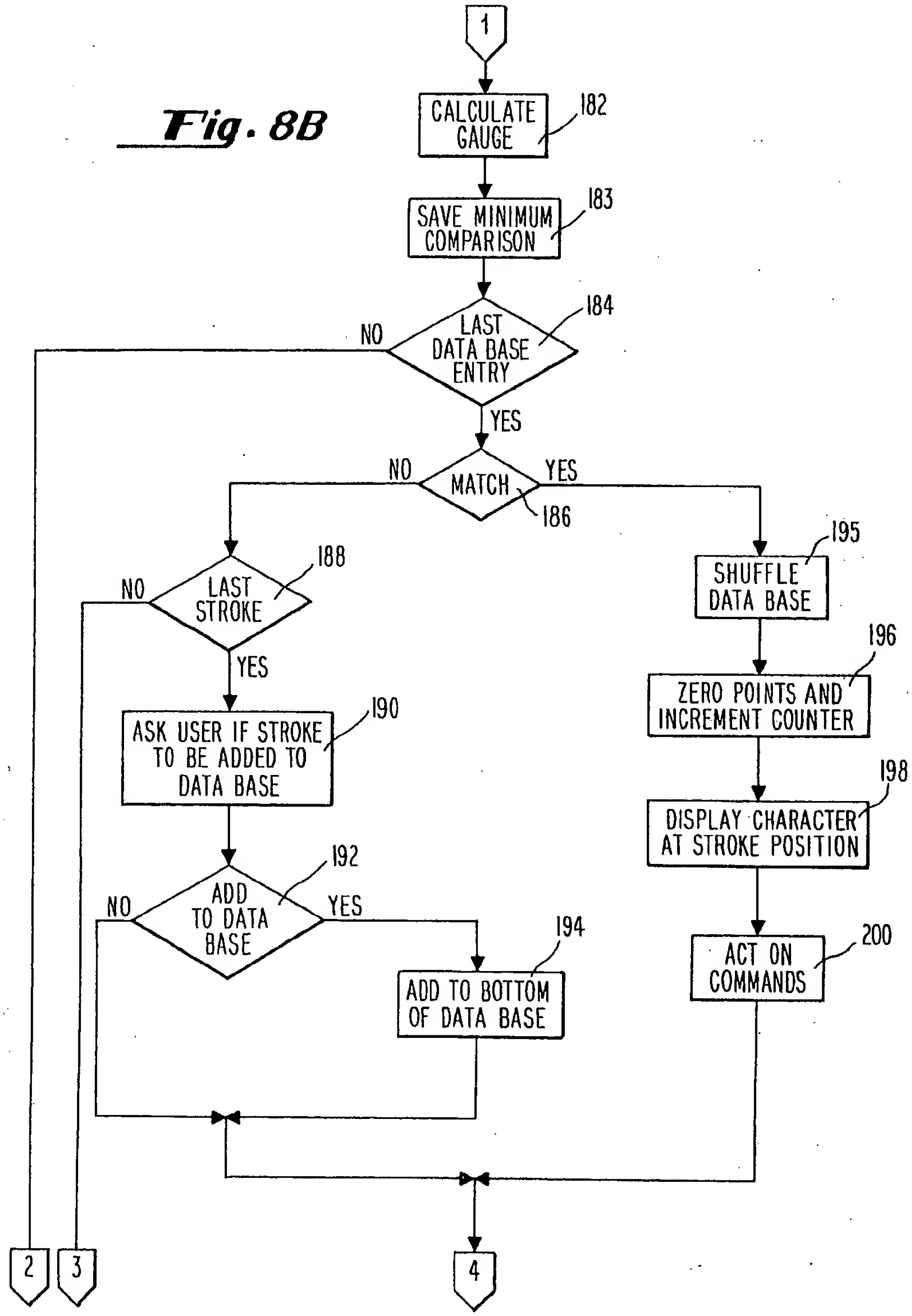
1
CALCULATE GAUGE
182
SAVE MINIMUM COMPARISON
183
LAST DATA BASE ENTRY
184
NO
YES
MATCH
186
NO
YES
LAST STROKE
188
NO
YES
ASK USER IF STROKE TO BE ADDED TO DATA BASE
190
ADD TO DATA BASE
192
NO
YES
ADD TO BOTTOM OF DATA BASE
194
SHUFFLE DATA BASE
195
ZERO POINTS AND INCREMENT COUNTER
196
DISPLAY CHARACTER AT STROKE POSITION
198
ACT ON COMMANDS
200
2
3
4

Fig. 8B

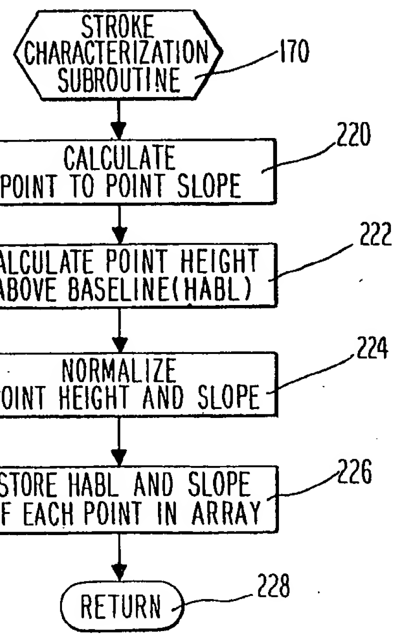
STROKE CHARACTERIZATION SUBROUTINE
170
CALCULATE POINT TO POINT SLOPE
220
CALCULATE POINT HEIGHT ABOVE BASELINE(HABL)
222
NORMALIZE POINT HEIGHT AND SLOPE
224
STORE HABL AND SLOPE OF EACH POINT IN ARRAY
226
RETURN
228

Fig. 9

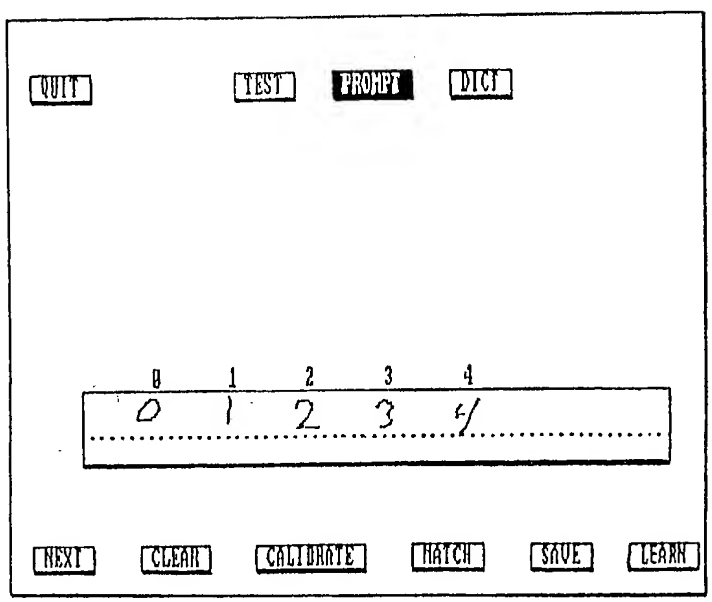
QUIT
TEST
PROMPT
DICT
0 1 2 3 4
NEXT
CLEAR
CALIBRATE
MATCH
SAVE
LEARN

Fig. 10

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back boards.

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we will be using CMOS technology through out. We also expect to use Surface Mounted Technology(SMT). SMT allows a denser packing, more shock resistence, and lower manufacturing costs, although initial costs may be slightly higher. The final decision on SMT will depend upon the availability of

Fig. 11A

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back boards.

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we

Fig. 11B

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back boards.

now is the time

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we

Fig. 11C

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back boards.

Now is the time

now is the time

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we

Fig. 11D

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back board. Now is the time

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we will be using CMOS technology through out. We also expect to use Surface Mounted Technology(SMT). SMT allows a denser packing, more shock resistence, and lower manufacturing costs, although initial costs may be slightly higher. The final decision on SMT will depend upon the availability of

Fig. 11E

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back board. ~~Now is the time~~

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we will be using CMOS technology through out. We also expect to use Surface Mounted Technology(SMT). SMT allows a denser packing, more shock resistence, and lower manufacturing costs, although initial costs may be slightly higher. The final decision on SMT will depend upon the availability of

Fig. 11F

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back board.

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we will be using CMOS technology through out. We also expect to use Surface Mounted Technology(SMT). SMT allows a denser packing, more shock resistence, and lower manufacturing costs, although initial costs may be slightly higher. The final decision on SMT will depend upon the availability of

Fig. 11G

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back board.

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Given our cost and power requirements we expect that we will be using CMOS technology through out. We also expect to use Surface Mounted Technology(SMT). SMT allows a denser packing, more shock resistence, and lower manufacturing costs, although initial costs may be slightly higher. The final decision on SMT will depend upon the availability of

Fig. 11H

I - Printed Circuit Board

The board's size, cost, power consumption, and reliability are critical to the sucess of our product.

The board must fit in our package, which limits its dimensions to something like 10" X 5", or about 50 square inches. Room must be available for dealer installable options such as additional memory and modem - possibily as piggy back board.

Power consumption must be low enough so that the battries in the product can provide for about 8 hours of operation and about 2 weeks of data retention. These are exclusive, i.e. 8 hours of operation followed by 2 weeks of data retention is required. With this kind of power consumption heat should not be a problem.

Fig. 11I

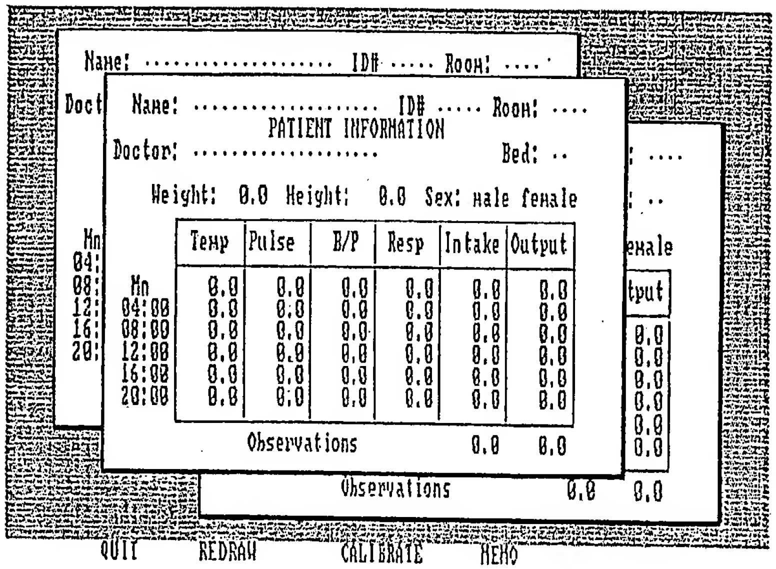

Fig. 12A

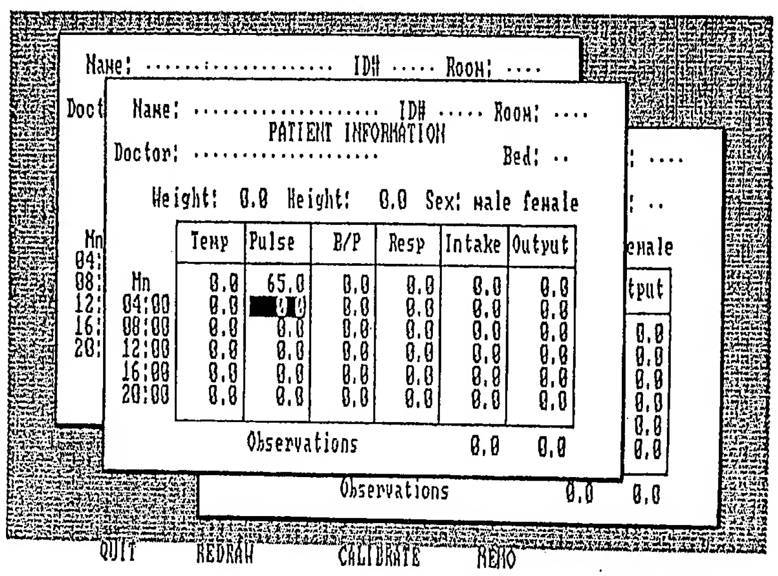

Fig. 12B

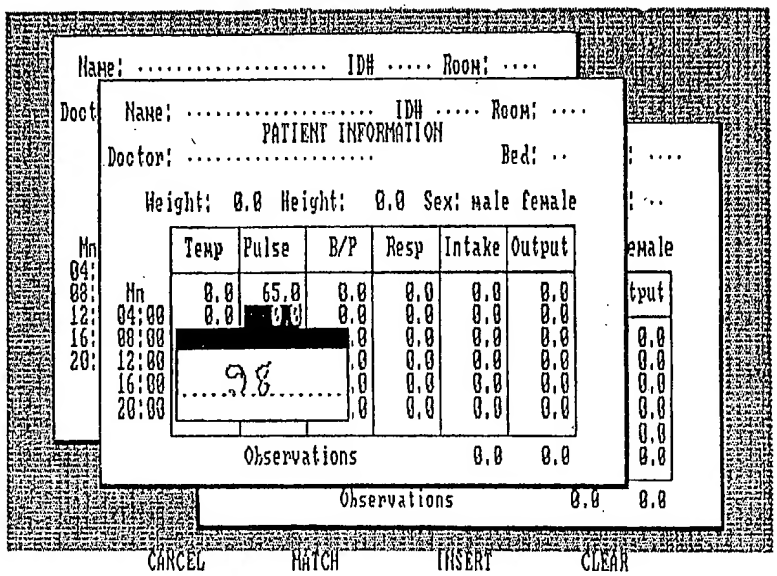

Fig. 12C

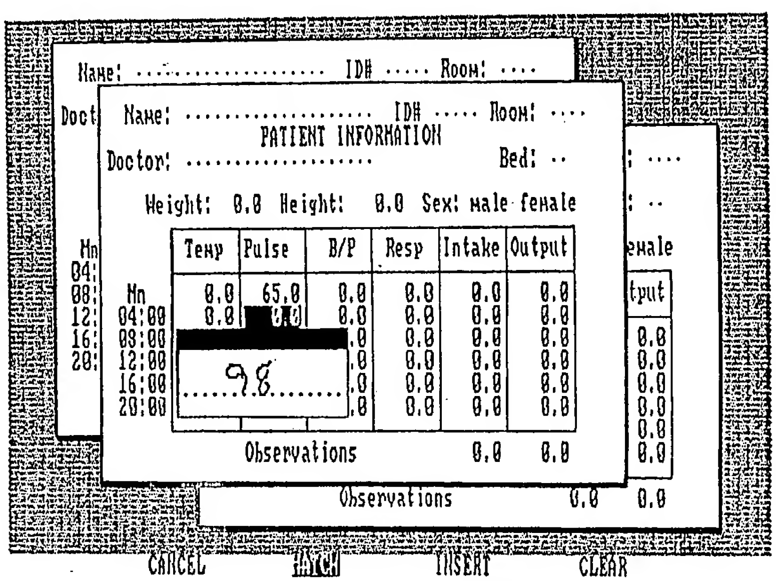

Fig. 12D

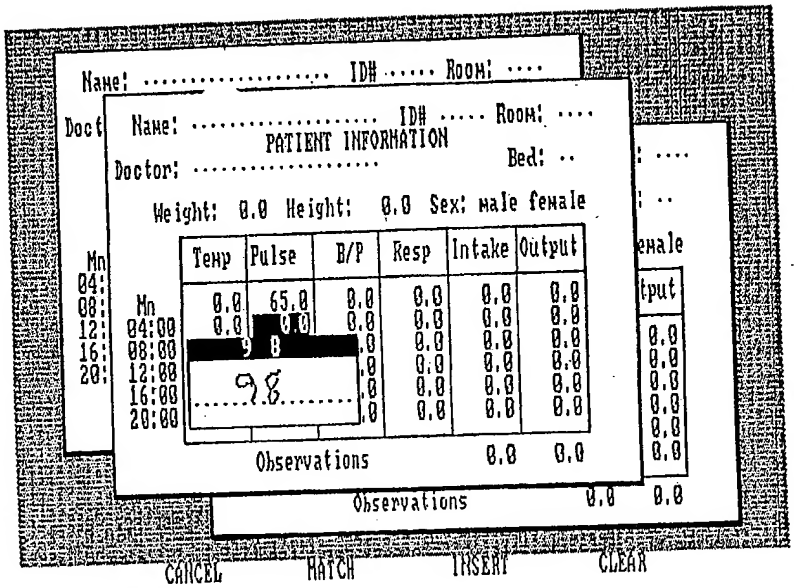

Fig.12E

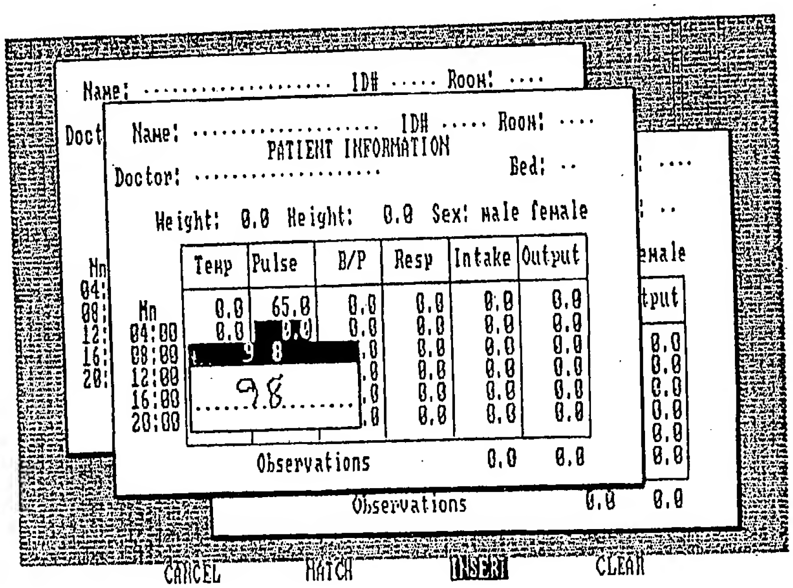

Fig.12F

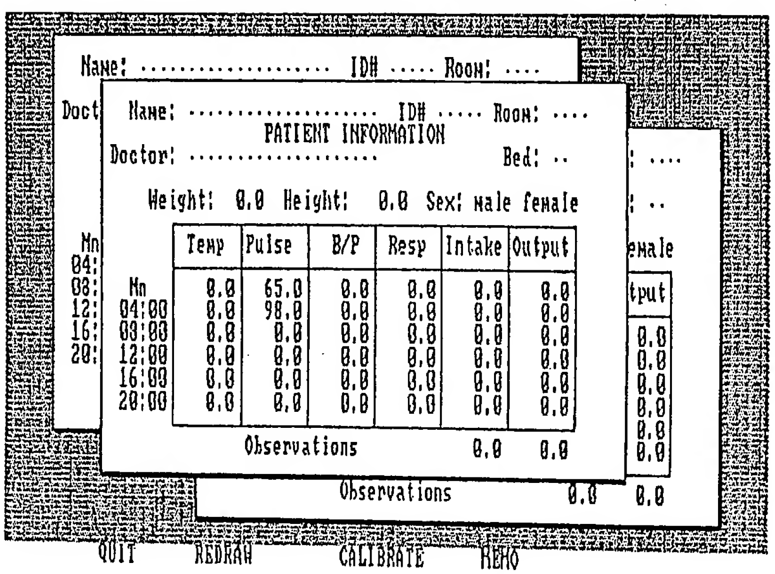
Name: .... ID# ..... Room: ....
PATIENT INFORMATION
Doctor: .... Bed: ..
Weight: 0.0 Height: 0.0 Sex: male female
Temp
Pulse
B/P
Resp
Intake
Output
Mn
04:00
08:00
12:00
16:00
20:00
0.0
65.0
98.0
Observations
QUIT
REDRAW
CALIBRATE
REMO

Fig. 12G

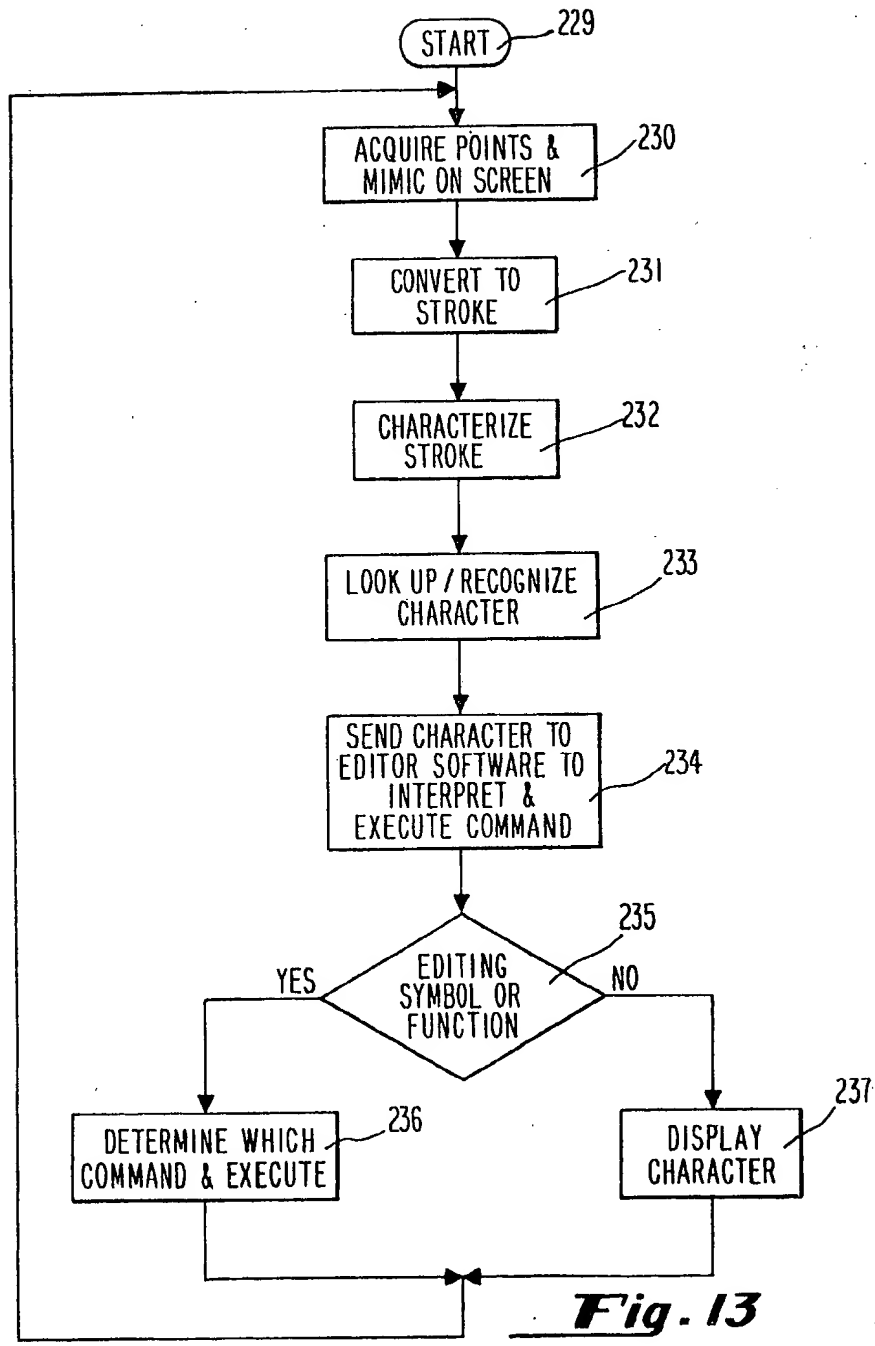
START
229
ACQUIRE POINTS & MIMIC ON SCREEN
230
CONVERT TO STROKE
231
CHARACTERIZE STROKE
232
LOOK UP / RECOGNIZE CHARACTER
233
SEND CHARACTER TO EDITOR SOFTWARE TO INTERPRET & EXECUTE COMMAND
234
EDITING SYMBOL OR FUNCTION
235
YES
NO
DETERMINE WHICH COMMAND & EXECUTE
236
DISPLAY CHARACTER
237

Fig. 13

SPECIFICATION

Handwritten keyboardless-entry computer system

The present invention relates generally to a keyboardless input system to a computer, and when combined with a central processing unit, to a keyboardless-entry computer system. More particularly, the present invention relates to an information storage, manipulation and transfer device on which text, data, computer commands and functions can be entered by writing alphanumeric or any other characters and symbols by hand with a penlike stylus on an Input/Output (I/O) screen. In a preferred embodiment the I/O screen includes a transparent touch screen incorporated over a substantially flat output display. The present invention in its preferred embodiment is a self-contained computer system but can also function as a peripheral to a host computer.

Large amounts of information and sophisticated applications software are now available on conventional keyboard computers. The utility of this information and of application software could be greatly increased if text and data could be entered and applications software manipulated by writing in a normal fashion directly on a flat display. Thus, there is a need to allow the utility of computer technology to be extended for use by non-keyboard oriented individuals. There is also a need for a portable computer system that is lightweight, reliable, accurate, inexpensive and permits use while standing or walking. One way to reduce expense and size and increase utility is to employ a keyboardless entry system, such as a touch screen. However, this type of input device does not easily allow accurate detailed input within a real time framework with high resolution in a manner which is familiar and natural to the user.

Many positioning technologies can be used to meet the requirements of the position sensing input technology. Essentially these requirements include accuracy, resolution and speed. The technologies include: mechanical, electrostatic, electromagnetic, acoustic, optical, and inertial. The desire in this system is to have its use as similar as possible to writing with pen or pencil on paper. One problem is proximity—a pen on paper only leaves a trail when actually in contact. Many of these technologies require an additional "pen down" sensor which is awkward to use in many commercial pens. Another problem is writing angle—a pen leaves the same trail independent of writing angle. Many of these technologies have the position detector displaced from the pen tip, so pen angle causes erroneous displacements. Beyond these general problems, each technology has numerous advantages and disadvantages in (1) the pen: size, weight, shape and whether it needs to be powered and/or wired, and (2) the writing surface: transparency, smoothness, "feel", and whether or not physical contact is needed (as opposed to pressure transmitted through overlaying sheets of paper).

A number of self-contained devices for viewing and processing large amounts of information are known. Most employ optical, magnetic or solid-state electronic storage means to store data. Illustrative of this body of art is U.S. Patent 4,159,417 to Rubincam which discloses a portable electronic book configured to provide selective page by page call-up of large amounts of digital data and displays it on a flat, solid-state screen. The preferred embodiment in the Rubincam patent uses an insertable holographic card, which may contain several hundred pages of text in digital form, as the main storage means. Rubincam's device, however, does not allow new information or text to be entered or manipulated.

In U.S. Patent 4,016,542 to Azure an electronic data collection system is disclosed which employs a solid state Random Access Memory (RAM) for its primary memory. This patent, which discloses a conventional keyboard for data entry, is directed to a hand-held portable data storage and transmission system, as well as an LED display and various Input/Output (I/O) connectors.

U.S. Patent 3,487,731 to Frank discloses a means of converting handwriting into character data through the use of a computer system. The disclosed invention is based on matrix pattern matching and does not employ any coincident display technology.

U.S. Patent No. 4,491,960 to Brown shows a handwritten symbol recognition system in which an array of image Points, in the form of a raster line sampling, is converted into segment-oriented lists which are filtered and compressed to obtain topologic features which are then analyzed with a logic tree decision mechanism.

U.S. Patent No 4,262,281 to Buckle *et al.* discloses a handwriting recognition device. The disclosed embodiment is for use with a host computer and does not employ coincident display technology.

U.S. Patent No. 4,475,239 to Van Raamsdonk discloses a text editing apparatus. The '239 patent cells for the use of paper as a medium for the entering of editing functions and requires a conventional keyboard for the inputting of text.

U.S. Patent 4,521,909 to Wang shows a dual level pattern recognition system. The system is designed for use with an optical instrument.

U.S. Patent 4,520,357 to Castleberry *et al.* discloses an electroscopic information display and entry system with writing stylus. The disclosed embodiment does not claim to have the speed

or accuracy to enable handwritten character recognition.

Additional prior art which discloses portable electronic devices that provide large amounts of various types of stored information include U.S. Patents 4,218,760 to Levy; 4,4115,486 to Laine; and 3,932,859 to Kriakides *et al.* The Levy and Kriakides *et al.* patents pertain to electronic dictionaries, while the Laine patent discloses a programmable television reminder system. None of these devices disclose the use of a handwritten input.

In U.S. Patents 4,071,691, 4,129,747, 4,198,539, 4,293,734, 4,302,011, 4,353,552, 4,371,746 and 4,430,917 to William Pepper, Jr. various methods or machine-human interfaces using finger touch are disclosed. The preferred embodiments in each of these inventions lack sufficient speed and resolution to allow handwritten character recognition with a stylus and are designed for other purposes. U.S. Patent 4,318,096 to Pepper teaches the use of a conductive stylus. The '096 patent pertains to graphic design and allows line width and line intensity to vary by applying pressure on the stylus with the results displayed on a conventional CRT screen. U.S. Patent 3,699,439 to Turner and U.S. Patent 4,055,726 to Turner *et al* discloses two methods for electronic position sensing through the use of a probe.

The invention in its various aspects is defined in the appended claims to which reference should now be made.

A preferred embodiment of the invention which is described in more detail below with reference to the drawings comprises a unique keyboardless computer system which has the ability to recognise and display Handwritten Symbols and cause the computer to display Font Symbols and, if desired, to execute editing functions pursuant to Editing Symbols, quickly, easily and at reasonable cost.

This preferred embodiment includes a computer housing with a flat display panel on which a user may "write" with a stylus, and has a capability to recognise Handwritten Symbols written on the panel with the Stylus and convert them to displayed Font Symbols, and/or to execute Editing Functions with Editing Symbols, all with a minimum of technical complexity for the user. Once the keyboardless, portable computer is loaded with the desired information and applications software, information and software can be used and responded to without requiring skills or knowledge related to state-of-the-art computers or other data source. The ease-of-use of the input technology of the apparatus enhances the utility of the computer for keyboard oriented individuals. Its portability also allows it to be used in applications and settings in which portable keyboard computers are awkward, difficult or impossible to use. For example, a multiplicity of blank, fully or partly completed forms may be stored in the portable computer memory. In a hospital, "sheets" of patient data can be stored in the memory of the portable computer, called up by a nurse as the nurse makes rounds and relevant data, such as blood pressure, temperature, etc., can then be entered manually with a stylus. These corrected or expanded forms can then be down-loaded into a central computer memory.

The requirements of the position sensing input technology are accuracy (Point to Point), resolution (absolute position) and speed (Points per unit time) to adequately define the written Stroke for recognition analysis. For the recognition apparatus and methods presently used, as described below, the present minimum requirements are: accuracy of 0.125mm (0.005 inch), resolution of 0.375mm (0.015 inch) and speed of 150 Points per second. This accuracy allows a 6mm (1/4") high writing line with over 10 raw input Points along a Stroke of a small letter. The resolution provides positioning of the symbol to within two pixels on a present display of 640 pixels to 225mm (9 inches). The speed permits about 50 raw input Points for a rapidly written single letter (1/3 second).

One embodiment of the present invention comprises a transparent input screen. As the user writes alphanumeric or other characters or symbols on the input screen, the character is represented as a stream of Points emulating written input with pen on paper. Once the discrete alphanumeric and other characters or symbols are complete, they are translated into computer text or computer commands that can be displayed on a display screen at a location that is preferably beneath the area on the input screen where they were entered. The embodiment also comprises a pattern recognition algorithm which allows the translation of any written character or symbol, such as ideographs and scientific symbols, into computer text.

In a particular, presently preferred embodiment, a keyboardless computer is configured as a manipulation and display device comprising a transparent touch screen and associated electronics placed over an 80 column by 25 line or larger display screen; a stylus for entry of data; a microprocessor and storage means; artificial intelligence/pattern recognition software and editing software; and a battery power system; and other I/O means.

As used herein, "Handwritten Symbols" are any symbols capable of being handwritten and having communicative effect. By way of example, and not limitation, numbers, letters, "Kanji" (Japanese ideograms) or other language symbols, editing symbols and engineering, scientific, architectural and mathematical symbols are Handwritten Symbols. Other examples of Handwritten Symbols are free-hand drawings or signatures or any other such written information uniquely configured by a particular writer. Handwritten Symbols may also include Editing Symbols (defined

below).

As used herein, "Font Symbols" are computer-generated symbols which are displayed in a predetermined font format. By way of example and not limitation, alphanumeric symbols may be Font Symbols and displayed in numerous font formats. Japanese or Chinese "ideograms" may also be Font Symbols, as may be engineering, scientific, mathematical, architectural or other such characters. Other examples of Font Symbols include any form which can be stored and displayed by a computer, e.g., a drawing of a car or a house.

As used herein, an "Editing Symbol" is a symbol (such as a caret, horizontal line, short vertical line, long vertical line, etc.) which is intended, when recognized, to cause the computer to execute a particular Editing Function (defined below), such as insert test (caret), delete text (horizontal line) delete a letter (short vertical line) or move a margin (long vertical line), to list a few representative examples.

"Editing Function" means any computer-generated text editing operation, such as by way of example and not limitation, insert text, delete text, move text and substitute text. Some primary Editing Functions are listed towards the end of the description below.

The preferred embodiment of a keyboardless entry computer system will now be described that includes a transparent input screen that generates positional information when contacted by a styles, and a display screen mounted physically below the input screen such that a character that is displayed can be seen below the input screen. The system includes a computer programmed to compile the positional information into Strokes, to calculate Stroke characteristics, and then compare the Stoke characteristics with those stored in a database in order to recognise the symbol drawn by the stylus. Key features of the system are: (A) transparent position sensing subsystem; (B) underlying display on which to mimic drawing of sensed positions and to show characters or symbols; (C) means to convert sensed positions first into plotted Points and then into recognised characters or symbols; and (D) means to "learn" to associate sensed input positions with a character or symbol. The apparatus can be housed in a portable unit having dimensions up to about 400mm by 400mm by 100mm (16"×16"×4") and a weight of up to about 7 kilograms (15 pounds), with a self-contained power supply.

Brief Description of the Drawings

The preferred embodiment of the invention will be described in more detail by way of example with reference to the drawings, in which:

Figure 1 is a schematic system block diagram of an embodiment of the present invention;

Figure 2 is a perspective view of the housing containing the operating element of the system;

Figure 2A is an enlarged portion of Fig. 2, with parts removed to show the positional relationship between the touch input screen and the display screen.

Figure 3 is a schematic diagrammatic view of the input screen, stylus and associated electronics;

Figure 4 is an overall schematic system block diagram of the apparatus of a keyboardless entry computer system according to the present invention;

Figure 5 is a schematic block diagram depicting the movement of data within the system when modified by handwritten characters and commands;

Figure 6 is an overall system block diagram depicting the hierarchy of software used to operate the system;

Figure 7 is a generalized block diagram of the character and pattern recognition algorithm.

Figures 8A and *8B* together are a detailed block diagram of the pattern recognition algorithm.

Figure 9 is a schematic block diagram of the Stroke characterization subroutine.

Figure 10 is a top plan view of a screen illustrating the "initializing" of the database for Handwritten Symbols.

Figures 11A through *11I* are a series of top plan views of screens depicting the operation of a text editing system.

Figures 12A through *12G* are a series of top plan views of screens depicting the operation of a data entry system.

Figure 13 is a generalized block diagram of the Linus Editor.

Detailed Description of the Preferred Embodiments:

With reference now to the figures, wherein like numerals indicate like elements throughout the several views, and in particular with reference to Fig. 1, an overall block diagram of a portable handwritten, keyboardless entry computer system 10 is depicted. The complete computer system is encased in a housing 12, indicated graphically by the dashed line, and includes a conventional general purpose digital microcomputer 14, described in greater detail hereinbelow. Input information is provided to microcomputer 14 by stylus 16 "writing" on writing or input screen 18. Stylus 16 (Fig. 2) is connected to the computer of system 10 with wire 17 (Fig. 2). As stylus 16 "writes" on input screen 18, a plurality of locating signals representative of a plurality of corresponding positional coordinates are transmitted to microcomputer 14. Microcom-

puter 14 has been programmed in accordance with a computer program described hereinbelow, to recognize the stream of locating signals and to store these signals in a computer memory. The programmed microcomputer 14 also provides a corresponding plurality of display signals to a display screen 20. Both input screen 18 and display screen 20 are described in greater detail hereinbelow.

Referring now to Fig. 2, there is shown a perspective view of keyboardless computer system 10 embodying the features of the present invention. Keyboardless computer system 10 is contained in housing 12, which is a rectangular enclosed casing having a sloped top surface 22 with a multi-line, solid state display area 24. Input screen 18 is depicted in Fig. 2A as being positioned over display screen 20. In this example, display screen 20 displays a plurality of horizontal lines 25 with the following indicia:

Name ____________________
Address ____________________
____

Handwritten entries are made above each line 25. The distance or space between two lines 25, denoted 26, is used by the system to normalize all distances, and lines 25 themselves serve as a reference axis or base line.

Below display area 24 on top surface 22 is a key input section 26 comprised of a plurality of "Softkeys" 28. Softkeys 28 can be programmed by the operator for any purpose, such as to enter computer commands. Exemplary commands for Softkeys 28 are "store," "recall," and "delete". In addition, Softkeys 28 can be used to switch between different programs or between modes (e.g. data entry mode and edit mode). However, Softkeys 28 are optional and are used to supplement the input obtained by handwriting the entries. Stylus 16, used for writing input data and commands in display area 24, also is used to activate the selected Softkey 28. An ON-OFF switch 30 is positioned on the side of housing 12 adjacent to Softkeys 28. A data output or peripheral connector 31 is located on the upper right side of housing 12.

Input screen 18 can be a conventional resistive type touch screen in which a voltage is applied to the screen edges and a stylus detects the voltage at the touched location. The writing surface is a transparent material, typically glass, coated with a thin, uniform, conductive layer (presently, vacuum deposited indium tin oxide). Vertical bus bars or conducting strips (not shown) are used along the two sides to apply the reference voltage to determine the "X" coordinates of the stylus position and horizontal bus bars or conducting strips (not shown) are used along the bottom and top to apply the reference voltage to determine the "Y" coordinates of the stylus position. In this embodiment, stylus 16 is merely an electric probe that, when physically in contact with the conductive layer, detects the local voltage at the Point of contact, which will vary with the distance from the conducting strips or bus bars. With the origin at the Point of voltage application, the X, Y coordinates are inversely proportional to the impressed voltage. Stylus 16 must make good contact to minimize adding resistance that would lower the voltage detected, and thus add an erroneous distance increment. In a presently preferred embodiment, a soft graphite tip is used. The voltage is conducted from the pen through a wire, such as wire 17 in Fig. 2, to an analog to digital converter for use in the computations described below. The stylus may be a charged "pen" as described herein, a light pen as is well known in the art, or any other hand-held device which can outline Handwritten Symbols on a screen.

An example of a conventional electrostatic screen is disclosed in the aforementioned 4,318,096 Pepper patent. This resistive type screen has the advantage that the interference caused by the user's hand touching the screen is minimized.

Both horizontal and vertical position sensing is provided by alternately switching the voltage impressed on a conductive layer between the pairs of horizontal and vertical bus bars by an interface and multiplexer controlled by a microcomputer or microcontroller. In one commercially available input or touch screen, the bus bars are broken into a series of short strips with diodes to prevent the horizontal strips from shorting out vertical strips and vice versa. This technique is used in a commercially available touch screen from Touch Technologies of Annapolis, MD. and Elographics of Oak Ridge, Tenn.

Referring now to Fig. 3, an alternate embodiment of a low power position sensing novel input screen 33 is described in greater detail. Input screen 33 is also for determining an X,Y position on an electrically resistive plate 34. A stylus 35 containing a voltage source, such as battery 36 or a voltage transmitted to stylus 35 from an external source such as the system power supply, is used to touch screen 34 and apply a voltage at the touched position. When the touched position is charged by stylus 35 with a positive voltage with respect to a plurality of plate measurement Points 37, the voltages at these Points will vary with the distance to the pen position, such as position X_1, Y_1, indicated at 38. These voltages are sequentially measured in the X and Y directions by using conventional means, such as disclosed in the aforementioned

prior art patents. In Fig. 3 these means are a conventional interface/multiplexer 42. A conventional Analog-to-Digital converter 43 converts the detected voltages into a digital signal. A microcontroller 44 receives the digital signal, performs standard checks to insure the signal's numerical value is "valid" (e.g. is within the possible range of voltages), and then converts the voltages to X and Y distances in the manner described herein. Microcontroller 44 is conventional, but could be replaced by a system computer. Microcontroller 44 provides a digital signal representative of the X and Y distances to measuring Point 38 to an output port 46. Port 46 can be a conventional RS 232 port. Alternatively, microcontroller 44 could translate Point X_1, Y_1 to any other reference Point, such as a Point on base line 25 (Fig. 2).

As long as there is no contact by stylus 35 at position 38 or any other position on plate 34, no current flows and power consumption is minimal. An incident measurement of the voltage at the measurement Points may occur by using ramped voltage at the positioning Point and timing when the measuring Point voltage exceeds a preset back voltage.

There are various options for the input or touch screens 18 and 33: resistive plate 34 or its equivalent for screen 18 can be transparent or translucent and the position Point can be made by a stylus or a finger of the user, or a connecting Point of an overlapping conductivr screen (such as the commercially available touch screens from Touch Technology, Annapolis, MD.). Input screens 18 and 33 can be a physical solid surface which is transparent or translucent and can be glass or plastic such as Mylar. The surface can be coated with a conductive/resistive substance like indium tin oxide. Other physical surfaces can use sound or electromagnetic radiation transmission from the touched position to a reference Point or Points and the distance is determined by the time delay or phase shift. Alternatively, input screens 18 and 33 can use an ethereal or geometric surface defined by an electromagnetic, optical or sonic field.

Position detection can be accomplished with electrical contact closure by resistive, capacitive or inductive coupling, remote sensing by sonic, electric or magnetic fields or by light (UV, IR, or microwave) scanning.

The advantages of the low power position sensing input arrangement over other such screens are: (1) it makes stand-by power requirements minimal; (2) it eliminates distortion due to opposing parallel "bus" bars in conventional touch screens; and (3) when a ramped voltage is employed, it eliminates the need for an A/D chip which is a major cost factor in state-of-the art touch screen technology.

The coefficient of friction of the screen 18 is desirably selected to be "rough" enough to offer some resistance to the movement of stylus 16 on the screen. If the screen were too smooth, the stylus would slide too easily and would be difficult to control.

Reference is now made to Fig. 4 which discloses an overall system block diagram of the major electronic circuitry used in the preferred embodiment of the present invention. Microcomputer 14 includes a microprocessor 50, interconnected to a plurality of other electronic elements by means of data path or bus 52. A Read-Only-Memory (ROM) 54 which is programmed with the operating and application programs and a battery powered Random Access Memory (RAM) 56 is connected for bidirectional data flow onto bus 52. Microprocessor 50 may be a conventional single-chip eight-bit or sixteen-bit device which functions to execute the fixed control programs residing in ROM 54, and further receives control programs from and provides control signals to the other electronic elements via bus 52. Microprocessor 50 may be of the type commercially designated Z80 (manufactured by Zilog Microcomputers of Cupertino, California), of a type 8088 device (manufactured by Intel Corp. of Santa Clara, California) or any similar or more powerful microprocessor. ROM 54 may be of the type 2564 or 4764, both manufactured by Texas Instruments of Dallas, Texas. The storage capacity of RAM 56 is determined in part by the sizes of the application programs, the operating program and the database. As discussed below, RAM 56 may be of the static SRAM or dynamic DRAM type. The primary requirements of RAM 56 are that it have sufficient storage capacity and that it require a minimum of input power.

A battery 58, such as a lithium battery, provides power for making the memory of RAM 56 non-volatile for extended periods of time. A battery pack 60 containing the well-known rechargeable types of batteries is used to provide the various voltage levels required by the other electronic elements of microcomputer 14.

Alternately, the storage function of RAM 56 may be served by a non-volatile device which requires no power for maintaining storage, such as an electronically erasable and reprogrammable memory (EEPROM), or devices using magnetic bubbles or capacitance. State-of-the-art disk or tape may also be used for mass storage. Suitable bubble memory devices include types 7110 and 7114 which have storage capacities of 1 megabit and 4 megabits respectively. (Both are manufactured by Intel Corp.). Furthermore, it is possible to use a single integrated circuit chip which includes microprocessor 50, at least part of ROM 54 and at least part of RAM 56.

Also connected to bus 52 is an EIA RS-232 serial interface 62 which provides a means for inputting and outputting data. Data is provided to bus 52, (usually to RAM 56) by interconnecting an external data source to RS-232 port 62 directly to the microprocessor 50 and other

elements of the microcomputer 14. Offloading data from RAM 56 can also be done by microprocessor 50 to an external computer, other data gathering device, a mass data storage device (e.g. floppy and hard disk drives) or an electronic telecommunications system. In like manner data can be communicated through port 62 to a printer (not shown) from interconnecting bus 52.

Stylus 16 is used to write on input screen 18 and to cause the generation of X,Y coordinate information by conventional touch screen interface electronics circuitry. The coordinate information is communicated via the bus 52 for control use by system 10. The solid state display 20 consisting of a multi-line display—illustratively 80 columns by 25 lines—is interconnected to bus 52 through a display interface 66. The fundamental requirements for the display are that it be substantially flat and sufficiently thin for use in the present invention. The display may be of the following types: scanning types such as a cathode ray tube, projected types such as a rear-view projector, light emitting array of Points types (e.g., electroluminescent or plasma discharge) and light blocking array of Points types (e.g., liquid crystal displays, solid state PLTZ or magneto-optical). In addition, it is preferable that the display be compatible with input screen 18 in size, configuration and transparency, and that both be low power consuming types.

The X,Y coordinates are input to keyboardless computer 14 via input screen interface electronics 64 and communicated via bus 52 to microprocessor 50 which executes programs stored in ROM 54 and RAM 56.

The number of Points (*i.e.*, sets of X,Y coordinates) used in defining each Handwritten Symbol and the speed at which Points are identified are important for practical utility. It is desirable to use at least about 4 Points per millimetre (100 Points per inch) and at least about 100 Points per second to define Handwritten Symbols. It is to be noted that the more Points per unit length that are identified, the greater the accuracy of the system in identifying Handwritten Symbols—however, more Points being identified will slow down the speed of identification and require more computer memory. Accordingly, a balance will have to be achieved, based on the size (available memory and processing ability) of the computer system and the requirement for speed of response and accuracy. For most purposes, standards in the range from about 4 Points per mm (100 Points per inch) and 100 Points per second to about 8 Points per mm (200 Points per inch) and 200 Points per second will be suitable.

It is also to be noted that the greater the precision of the system in identifying the X,Y coordinates of each Point the fewer the number of Points needed to be identified per unit length and per second to accurately identify Handwritten Symbols. Conversely the less the accuracy, the more Points that are needed.

Point resolution is needed to place Points where intended, *e.g.*, to write an editing symbol precisely between the two characters. Ideally, resolution to a single display pixel is desirable. However, operationally, resolution within two displayed pixels is sufficient for a display with 640 pixels in a 225mm (nine inch) horizontal scanline.

When switch 30 (Fig. 2) is positioned to "power on", the basic display mode is activated and microcomputer 14 (Fig. 4) programmed by the operating system, causes a menu to be displayed on display screen 20 (Fig. 1). The menu presents various software options. A primary software function, editing, functions in a manner similar to conventional word processing software with the difference being that handwritten characters, symbols and commands are interpreted by the system as if they were entered from a conventional keyboard. The system is capable of learning the editing symbols used by a particular writer for functions such as indent, insert, delete, move and reformat and translates those symbols into digital command functions. Optionally, Softkeys 28 (Fig. 2), activated by touching those areas on the input screen with stylus 16, function like conventional hard function keys on a computer keyboard.

The system is particularly adapted for use as an interactive screen editor or word processor. After a writer retrieves a document by (for example) touching the displayed name of an existing file with the stylus or by writing the name of the file on the screen, all usual editing functions can be performed with stylus entry. When the user wishes to change a displayed character or symbol, he may simply write over the displayed character or symbol and as described hereinbelow the pattern recognition algorithm will translate the written entry into computer text. For example, the editing software allows text to be eliminated by simply drawing a line through it and a conventional caret symbol may be used to change the operating mode to the insert mode. In the insert mode, display screen 20 provides additional space for entry of handwritten characters or symbols which are inserted in the text after the Point where a caret was written in. Text can be moved simply by placing brackets or other user-defined delimiters around a displayed phrase or word and writing a caret or other user-defined symbol in the area of the text in which the user wishes this material to appear. New margins can be set by drawing vertical lines down the side of the displayed text where the new margins should appear.

The basic editor software also allows new documents to be created by simply writing Handwritten Symbols on the screen. All documents can be stored, changed and communicated in the manner in which these functions are accomplished on a conventional word processing

system with the difference that these functions are accomplished with handwritten Editing Symbols on the (optional) screen or by touching the Softkeys with the stylus. The composite text thus produced and stored can be subsequently offloaded through the RS 232 port 62 (Fig. 4) to another computer, a similar device, an external data gathering device or recording device, into a printer or through a telecommunications system.

In addition to these major operating modes, a number of ancillary elements and features add to the utility of the system. A conventional alphanumeric keyboard (not shown) containing a full set of keys can be connected to a conventional keyboard interface (not shown) to support the entry of alphanumeric characters. An AC/DC power connector may also be used in those applications when portability is not needed and when needed to meet the power requirements of screen technologies such as gas plasma displays and electroluminescent displays.

In actual use the keyboardless computer can function in any applicxation or environment in which handwritten input translated into computer text is useful or necessary. Illustratively, the device can function as a new generation word processor, or for use in fields such as sales, nursing, inventory control, census taking, claims adjusting, to name just a few of its many uses or as a general learning and testing device in education. Since the pattern recognition software can learn and translate into computer text from languages which are not made up of a small or limited set of alphanumeric characters (e.g., Japanese, Korean, and Chinese), it has particular utility for word processing and telecommunications in these languages.

In practice, it is particularly desirable to use a single computer screen to display any initial forms, Font Symbols or other displays to be edited, and to create a nearby "window" of blank space where Handwritten Symbols are to be written, displayed and identified, and where the Font Symbols corresponding to the Handwritten Symbols are to be displayed. In this way, the user can view the text being edited and the proposed insert or change without significant movement (if any) of the head and eyes. This is illustrated in Figs. 11A to 11D. This feature (proximity on the one screen of text to be edited and the window into which new text is to be handwritten) is very important to the simple, rapid, comfortable use of the system.

In a preferred embodiment of this invention, the system "learns" the handwriting of a particular user prior to actual use. For example, if using the Roman alphabet, the twenty-six letters of the alphabet and the numerals from 0 to 9 would be inserted into the database. Punctuation symbols, such as periods, commas, question marks, colons, semi-colons, hyphens and the like could also be inserted. There is virtually no limit to the Handwritten Symbols which can be recognized and stored in the database. Of course, the computer will have to store a suitable array of Font Symbols for conversion of the Handwritten Symbols. Different sets of Font Symbols could be created and stored in the permanent memory of the computer, as in ROM chip 54. For example, in English language usage, a chip could contain one (or more) fonts of numbers and letters, suitable punctuation symbols and appropriate mathematical symbols. Other chips could have stored Font Symbols for the Arabic, Cyrillic or Greek alphabets, Japanese, Chinese or Korean "Konji", symbols for use by architects or engineers, or chemical symbols (*e.g.*, benzene rings and the like).

In Fig. 10, one of a series of learning screens is displayed and the user is prompted to write the numbers 0 through 4. The computer will attempt to match the written numbers with the existing database (if any). If it cannot be matched because there is no existing database or because there is a poor match with an existing database, the character is added to the database. This learning process continues until all of the alphanumeric (or other) characters and symbols to be used are entered into the database. The system has the capability of storing multiple Stroke characterization databases for systems used by more than one user. The existence of a unique Stroke characterization database for each user has the further advanage of making the writing angle irrelevant. As a result, the system is adaptable to all handwriting styles and is usable by right-handed and left-handed persons. One feature may desirably be incorporated into the apparatus to accommodate left-handed and right-handed persons. This feature is a receptacle (not shown) for the stylus connector on either side of housing 12, so that the stylus 16 may be connected on the left side for left-handed persons and on the right side for right-handed persons.

Fig. 10 also provides an example of the use of "Softkeys". In addition to the input line, a variety of Softkeys appear. Each Softkey corresponds to a function that can be performed by the system. In order to execute the function, the user merely touches the indicated Point with the pen. The Softkey will then appear in reverse video and the selected function is performed. There are numerous advantages to Softkeys over traditional function keys. Some of the more significant of these are that the user is no longer required to memorize what function key performs what function; the need for keyboard overlays is eliminated: and different Softkeys can be made available (displayed and made operational) at different Points within a program.

Figs. 11A to 11I demonstrates some of the simplifications in word processing made possible through the use of this invention. In Fig. 11A a standard screen of text is displayed. The user of the keyboardless entry system decides that additional information needs to be added and draws

an insert symbol (e.g., caret) on the screen at the desired position. A data entry "window" then appears. (Fig. 11B). The text is written in as Handwritten Symbols (Fig. 11C), matched (converted to Font Symbols) (Fig. 11D), and then inserted (Fig. 11E). The operator reconsiders the addition and draws a horizontal line through the new material. (Fig. 11F). It is immediately erased (Fig. 11G). Next, the operator decides that a larger right-hand margin would be more appropriate for the text. A vertical line is drawn on the screen (Fig. 11H) and the margin is automatically adjusted (Fig. 11I).

A generalized block diagram of the editing process is provided in Fig. 13 and a description of that figure appears hereinbelow.

Figs. 12A–12G illustrate how a blank form may be used for a hospital patient. The user of the system first calls up the proper blank form (Fig. 12A). This may be done, for example, by touching an appropriate Softkey. The area where the information, in this case a pulse observation, is to be inserted is touched with the pen (Fig. 12B). After the desired location is highlighted, a "window" appears directly below the space where the observation is to be recorded (Fig. 12C). The nurse then touches the pen on the match box which appears highlighted (Fig. 12D). The software then matches the handwritten input to the corresponding Font Symbols and displayes the result (Fig. 12E). If there is an accurate match, the "insert" block is touched (Fig. 12F), and the new observation is added to the patient's records (Fig. 12G). This mechanism is clearly applicable to a wide variety of "blank forms" in which data is inserted into a form or corrected. For example, it could be used to correct or update financial information in a spreadsheet problem. Other information can be recorded in the same manner.

The reason for using a black background and white letters for the newly entered Font Symbols is to facilitate checking the accuracy of the inputted character. Although this is preferable, it is not essential and a while background and black letters is also acceptable.

The ability to create a window and input data on the same screen and in physical proximity to the text being edited or space for data to be input is an important feature of the apparatus, for it permits ease and speed in the use of the apparatus. The user's eye may focus on the space where the data will be inserted and the ability to contemporaneously display Handwritten Symbols and the corresponding Font Symbols makes it easy to see errors, when the system "misreads" a Handwritten Symbol, and then correct errors quickly and easily.

Referring first to Fig. 5, the overall operation and functioning of the pattern recognition software will now be described. When the operating system calls the pattern recognition program, the program begins in terminal 75 where a number of variables and counters are initialized. The software then proceeds to decision diamond 76 where the program determines if stylus 16 (Fig. 2) is in contact with input screen 18 (Fig. 2A). The system provides a "pen down" signal, as shown in processing box 78, as well as the X,Y coordinate voltages as locating signals, as described above. Microcomputer 14 (Fig. 4) converts the X,Y coordinate locating signals into Stroke characteristics using programs stored in ROM 54 (Fig. 4), or a separate microcomputer can do the conversion, such as microcontroller 44. If a pen down signal is received, the software proceeds to processing box 80 where the individual locating signals are combined into "Strokes," a stroke being defined as the Point locating signals produced between a "pen down" signal and a "pen up" signal.

The system then calculates a transform, as described below, for each Point, transforming the Point coordinates from the X,Y cartesian coordinate system to a relational coordinate system. The software next proceeds to processing box 82, where it compares the Stroke with previously entered Strokes accumulated into a database, and determines if the Stroke is represented by a symbol in the database. If a match is found (if the Font Symbol represented by the Strokes is recognized), as indicated in decision diamond 84, microprocessor 50 (Fig. 4) causes the symbol to be sent to display screen 20 (Fig. 4) as indicated in processing box 86. If a match is not found, microprocessor 50 (Fig. 4) causes a message to be displayed, as indicated in processing box 88, which requests further input from stylus on input screen 18 (Fig. 4) by either flashing an entry which is close to a match or a non-recognition symbol.

As mentioned above, the software compares the Stroke characteristics of each Handwritten Symbol to data entries previously stored in a database. In a preferred embodiment, the database is arranged into sections of characters or symbols by the number of Strokes needed to make the character or symbol. Within each section, the entries are randomly arranged at first, but after use, as explained herein, the most frequently used entries "rise" to the top of the database. It should be noted that each user will have his or her own particular style of writing a Handwritten Symbol and that each Handwritten Symbol may have a number of different variations.

For example, many people write the lower case letter "h" using a single Stroke. They do this by starting the pen on the writing tablet at a Point where they wish to place the top of the letter, drawing a vertical line downwardly to the base line, then without removing the pen from the paper, proceeding back up to the midPoint of the previously drawn vertical, line, over to the right and down to the base line when the pen is picked up from the paper. On the other hand,

these same people may draw the upper case letter "H" using two Strokes. They do this by drawing the left hand vertical line and horizontal line as is done for the lower case "h", picking the pen up from the tablet, and then drawing the right hand vertical line. Appendix I displays the data of the Stroke data Points for these two letters as the data is stored in memory after having been generated by an embodiment of the present invention.

As shown in Appendix I, the letter "h" as drawn at one particular time by one user has one Stroke (ns=1) with 20 Points (np=20) and x and y coordinate characteristics for the minimum, mean and maximum normalized values (1/80th of a line width) as follows: −17 and −6; 0 and 18; and 19 and 60, respectively. The values in the first vertical column are the Point-to-Point slopes, normalized to 360°/256. The values in the second vertical column are the Point-to-Point average vertical positions above the base line, normalized to 1/80 of the line width. A typical line width is about 1mm (0.04 inches).

Referring now to Fig. 6, a software hierarchy of programs is depicted. At the top, overseeing the entire operation of computer system 10 (Fig. 1), is an operating system as indicated by box 90. Applications programs shown in boxes 92 and 94, residing in RAM 56 (Fig. 4) and ROM 54 (Fig. 4) can be executed by microprocessor 50 (Fig. 4) under control of the operating systems. When a Handwritten Character is required or is indicated by an interrupt, handwriting recognition software 96 is called. A first subroutine, indicated in box 98, encodes the X,Y coordinates into Strokes. The characteristics of the Strokes are then defined by a subroutine 100 followed by comparisons of the Strokes with a database that has been loaded from ROM 54 (Fig. 4) into RAM 56 (Fig. 4). The comparison is made by a subroutine 102. When the operating system is in the "learning" mode, the database is updated with the new Stroke data and symbols, as indicated in box 104. Similarly, a previously stored document can be edited by applications program 92 by using edit function 94 as called by the operator, who provides the instructions as input using the subroutines 98, 100 and 102 of handwriting recognition program 92.

Referring now also to Fig. 7, operating system 90 (Fig. 6) executes the Handwritten Character recognition software 96 (Fig. 6) by accepting as input the X,Y coordinate Points, depicted in box 110, of the position of stylus 16 (Fig. 2) on input screen 18 (Fig. 2) and encodes these Points into Strokes as depicted in box 112. The program then characterizes the Strokes by some description set, such as considering the length, curvature, slope, and position of the Stroke, as depicted in box 114. In box 116 the best comparison is then found of the characterized Stroke or sequence of Strokes with those in the database. If a sufficiently close match is found, the character is identified in box 118 and the database entry is swapped with the entry above it as shown in box 120. In this way, the most frequently identified characters will "rise" to the top of the database and the overall system performance, as measured in time to find a match will be increased. If a match is not found, the user can add to the bottom of the database, as indicated in box 122.

With reference now to Figs. 8 and 9, a flowchart of the computer program to recognize a particular Stroke sequence is set forth. The computer program begins in terminal 150 and proceeds to process the X,Y voltages from processing box 152, the voltages having been converted to a digital signal. The program then proceeds to decision box 154 where the program determines whether the pen or stylus 16 (Fig. 2) is out of contact with input screen 18. This determination is made by both the X voltage and the Y voltage being zero. If the program determines that the pen is up, then the Stroke is determined as having been completed and the program branches to decision box 156. In decision box 156, the program determines whether there are less than three Points in the Stroke and if so the program branches to decision box 158. In decision box 158, the program determines whether there are zero Points in the Stroke. If there are zero Points in the Stroke, then the program loops back to the beginning of processing box 152 where another set of Points is read. If the Point counter (incremented in processing box 164) indicates that there are more than zero Points, the program branches to processing box 172. In processing box 172 the Stroke is identified as a dot and its height above the base line (HABL) is calculated in processing box 173. From processing box 173 the program proceeds to processing box 171.

However, if the pen down signal is received, the program branches to processing box 160 where the voltages are scaled to determine the coordinate Point using the following formulas:

$$X = a_1v_1 + b_1$$

$$Y = a_2v_2 + b_2.$$

The constants a_1 and b_1 are scaling parameters that are determined from calibrating the input surface of the particular display.

Once the voltages are scaled, the program proceeds to decision diamond 162 where the program determines whether it is an erroneous Point. This is done by comparing the distance between Points and eliminating a Point if the distance is to great (greater than 2.5mm or 0.10

inches is presently used). On the other hand, a Point is also eliminated if the Points are too close together. Points are presently thinned out if they are within about 0.4mm (0.015 inches).

The comparison problem that exists for the first Point is resolved by determining if a Point is the first Point after a pen is down and then that Point is used only to check the next Point which is accepted, assuming that that Point is within the maximum distance of 2.5mm (0.10 inches).

If the distance between Points is determined as being outside the two criteria, the program drops the Point and branches back to the top of processing box 152 to read another pair of coordinate Point voltages.

On the other hand, if the Points fall within the criteria, the program continues to processing box 164 where a Point counter is incremented to keep track of the number of Points. This number is used in decision diamond 156, as mentioned hereinabove. The program then continues to processing box 166 where the Points are smoothed according to any one of a number of formulae. Smoothing is used to minimize noise from digitization, from erratic hand motion and from electronic noise. The simplest smoothing technique is a multiple Point average which results in calculating new Points $(x_j'y_j')$ as follows:

$$x_j' = \sum_{i=n_1}^{n_2} x_i/(n_2 - n_1 + 1).$$

And similarly for y_j' smoothed over Points $n_1 - n_2$.

Another simple method is called the running weighted average method and utilizes the following formula:

$$x_j' = \alpha x_i + (1-\alpha)x_{j-1}'$$

Alpha is a weighting constant that is usually positive (and less than one) and has been used at 0.25. The summations have been taken with n_2 minus n_1 equal to one. A third method involves what is called a spline fit wherein the following formula is used:

$$x_i' = (x_{i-1} + 4x_i + x_{i+1})$$

Any of the foregoing methods can be applied either before or after filtering. The filtering is done so as to reduce the number of input Points and to space data so that difference and/or angle calculations can be made within acceptable random error bounds. A simple process of thinning a sequence of Points by excluding the acceptance of subsequent Points within a set distance of the previously accepted Points has been found to be an effective filter.

From processing box 166, the program proceeds to processing box 168 where the Point is stored in an array that is incremented for each new Point since the last pen down signal. Thus, an addressable array of Points is created for each sequence of Points obtained from a pen down to a pen up signal. This sequence of Points is called a Stroke. From processing box 168, the program loops back to the top of processing box 152 where another Point is obtained until a pen up signal ends the Stroke.

In decision diamond 156, a determination was made as to whether there were less than three Points in a Stroke. By definition, if there are three or more Points in a Stroke, the Stroke is a line and not a dot. If there are three or more Points in the Stroke, the program branches to subroutine box 170. In subroutine box 170, discussed in greater detail hereinbelow with respect to Fig. 9, the Stroke is characterized as to its slope and base line height.

As can be seen from the foregoing, the segmentation of the stream of coordinate Points into a Stroke is based primarily on the determining when stylus 16 is "up" or not in contact with the surface of input screen 18. Alternatively, a stream of Points can be segmented to form Strokes on the basis of other considerations. For example they can be segmented based upon changes in a locally calculated curvature or upon a large local curvature. Local curvature is calculated by the change in distance along the input coordinates divided into the change in slope. This produces radius of curvature. When the radius of curvature changes rapidly with respect to distance along the input coordinates, or if the radius is too small, then a segmentation Stroke is assumed to end, and a new Stroke begun. Further segmentation techniques can look at the relative maximum and minimum in one or both coordinates and/or the curve crossings in the coordinates. However, these latter two methods have been determined to be less effective.

Characterizing a Stroke reduces the sequence of coordinates defining the Stroke or segment to a set of characteristics that are unique, generalized and minimal. Uniqueness refers to both

factors that the same characteristics are generated by the same coordinates and that the characteristics are sufficient to regenerate an approximation to the original coordinate sequence. The term "generalized" is used to mean that the characterization is invariant under such transformations so that the symbols are invariant (e.g., translation and scaling or stretching or small tilt). The scaling of all distances is accomplished by taking a ratio of the distance to a writing entry line width.

The minimal set of segment characteristics have the following features:

(1) Stroke position: one or more of centroid/average, extent extreme or beginning and ending Points determined relative to the writing entry line, to previous Strokes, or to character extent or center;

(2) Stroke shape is characterized by one or more of average slope, change in slope (which is a measure of average curvature) and/or a change in curvature, by sequence of slopes over specific length segments or over fractional lengths, or by a gross description of linear direction or circular completion and operating direction;

(3) And Stroke length as characterized by distance along the curve and/or the extent extremum along the coordinate system.

In one embodiment of the present invention, positioning by centroid, extent extremum, and starting and ending coordinates have been successfully used. The Stroke shape is encoded as a sequence of slopes and vertical positions (relative to Stroke centroid). The Stroke length can be approximated by the number of filtered Points. Alternatively, the average curvature can be encoded in total slope change (along with length), change in starting to ending slope or fitting the slope angle versus length curve for rate of change of slope angle. Additional characteristics that could be used include location of coordinate relative extrema, curve crossing, cusps, and Stroke direction. A particular method used to determine the unique characteristics is set forth below.

1. The numerical values of the Criteria for each Stroke of the Handwritten Symbol are determined.

2. The database values for each stroke of the previously learned Handwritten Symbol is determined and subtracted from the newly determined values respectively.

3. The absolute values of each difference are scaled, to make each of the five measurements reasonably equivalent to the others such as lengths scaled to height between lines.

4. The five thus-determined values are added.

5. A predetermined threshold is used as "goodness" test of recognition—too high a value and Font Symbols are infrequently recognized and too low a value causes Font Symbols to be misidentified. Thresholds of approximately 1,000 are used initially and then switched to approximately 100 for improved recognition. If the threshold is exceeded, the comparison is discarded and an error message is created and displayed.

6. The database is searched to find a numerical minimum difference. If the minimum difference is below the acceptable threshold for recognition, the corresponding Font Symbol is displayed on the screen or the command is performed, as the case may be.

It has also been found that the preferred classification of a Stroke is a continuous one, rather than one that is grossly discrete. For example, determining a slope by angle in 256 directions rather than in 8 is desirable. Other non-continuous classifications can include bars/arches/hooks, number and closure of cusps or horizontal or vertical Strokes.

From subroutine 170, the program proceeds to processing box 171 where both an individual Stroke and one or more preceding Strokes are compared with a database entry that is stored in RAM 56 (Fig. 4).

This comparison initially begins with three eliminating questions that are asked by the program in decision diamonds 174, 176, and 178. In each case, if the database entry is eliminated, the program proceeds to a processing box 180 where the address of the next data entry is received and from which the program loops back to the top of processing box 171. In decision diamond 174, the first eliminator is asked by seeking if the number of Strokes are different. If the number of Strokes are the same, the program proceeds to decision diamond 176 where the average Height Above Base Line (HABL) is calculated and compared with the HABL of the data entry. The entry is eliminated if the difference in the average HABL's is greater than one-half the height of the entry line. From a negative determination in decision diamond 176, the program proceeds to decision diamond 178 where the number of Points per Stroke are compared and the database entry is eliminated if the difference in number of Points is greater than ten. This determination varies from that made in decision diamond 174 because it is concerned only with the number of Points for each Stroke. However, in decision diamond 174, certain letters, such as the capital letters "E" and "A", have more than one Stroke per letter.

If a data entry is not eliminated by decision diamond 178, then the program proceeds to processing box 182 where the program calculates a gauge to be used to determine the closeness of the match between the selected entry in the database and the drawn Stroke. A presently preferred gauge is the sum of the absolute values of the differences between the

Stroke values and the database entry values of:

(a) distances or lengths n units of 1/80th of the line height (e.g., space 26, Fig. 2); and

(b) the slopes in units of 1/256th of 360° over all the Points along the diagonal of the comparison matrix.

Alternatively, Dynamic Programming Techniques can be used to optimize the comparison using off-diagonal elements as well.

From processing box 182, the program proceeds to decision diamond 186 where a match is determined. In actuality, a match is determined by the application of an arbitrary gauge (maximum allowable variance), which is the sum of absolute values of the differences between the entered Stroke characterization and that of the stored database entry. In processing box 183, the lower of the present gauge and the previous lower gauge is saved best match. The program then goes to decision diamond 184 where a determination is made whether the present entry is the last database entry. If it is not, the program branches to processing box 180 where the next entry is selected. If it is the last entry, the program proceeds to decision diamond 185 where a determination of a match is made on the basis of the gauge being below a predetermined threshold. This threshold is set by the user based on experience with the system.

If no match is obtained, the program branches to decision diamond 188 where a determination is made whether all Strokes have been checked. If the last Stroke has been checked, then the present Stroke is compared in sequence with a previous Stroke to all two Stroke entries. As in the comparison with all one Stroke dictionary entries, the best fit comparison for all entered Strokes is the recognized symbol or sequence of symbols.

However, if the last Stroke has been read and there still is not a match, then the program proceeds to processing box 190 where a question is displayed on display screen 20 asking the user if a new Font Symbol should be added to the database. The user responds and that response is used in decision diamond 192. Either the Stroke sequence is added to the database in processing box 194 and the program branches back to the top of processing box 152, or the program branches immediately to the top of processing box 152.

On the other hand, if a match is determined in decision diamond 186, the program branches to processing box 195 where the program shuffles the database by interchanging the serial location of the matched entry with the entry above it. The program then proceeds to processing box 196 where the program zeros the Point counter and the increment counter. The program next proceeds to processing box 198 where the matched and characterized Stroke or Strokes are displayed by the computer as the identified Font Symbol. This display is located at the position in which the entry was made on input screen 18 (Fig. 2).

From processing box 198, the program proceeds to processing box 200 where the program can act on any commands which it has interpreted. An alternative characterization of the Stroke uses the Points themselves rather than the length, scope, curvature and position.

With reference now to Fig. 9, this Stroke characterization is depicted in greater detail. Stroke characterization subroutine 170 essentially performs a mathematical transformation of each Point on a Point-by-Point basis to transform the Points from an X,Y Cartesian coordinate system to one in which the coordinates are the normalized slope of each Point and the normalized height of each Point above the base line (HABL).

Subroutine 170 first calculates the Point to Point slope in processing box 220 and then calculates the height of each Point above the base line in processing box 222. The slope and HABL of each Point are then normalized respectively to 1/256th of 2 Pi and to 1/80th of the width of the entry line in processing box 224. From processing box 224, the system proceeds to processing box 226 where the calculated normalized values for each Point are stored in an addressable array. The subroutine then returns to the program through terminal 228.

When the comparison is made between each Stroke and the stored values, the comparison is made by the normalized Point slope and Point height above base line. As mentioned above, a match is determined by an arbitrary gauge which is the sum of absolute values of the differences between the written Stroke and the stored or dictionary Stroke. The system learns by adding new Strokes to the dictionary database. Once the database fills up, those Font Symbols that are infrequently used are replaced by new entries.

In a working embodiment of the present invention, the algorithm successfully identified upper and lower case letters and numbers when written discretely from one another. For Handwritten Symbols that are written such that they are continuous, direct extrapolation would require searching a database sequentially for one, two, three, etc. Stroke symbols and looking for the best fit. Upon identification of a Stroke fit, a "new" letter is tenatively recognized, except that the next few Strokes are analyzed to check if they change the previous symbol for a better fit. For example, two Strokes that have been identified as "ones" would be combined and changed to the capital letter "H" once a cross bar was identified.

The system design demonstrated by Figs. 7 to 9 could easily be coded by one with ordinary skill in the art of computer programming into almost any computer language. The source code listings for one application program utilizing the disclosed invention is included as Appendix II.

The software in Appendix II is written in Microsoft Basic, a common computer language available for virtually all microcomputers and operating systems. The program is a complete text editing demonstration, which shows the improvements that can be made upon traditional word processing systems.

Program lines 2600 to 4000 contain the character recognition subroutine which includes the software code necessary to get X and Y coordinates. This section of the program corresponds to Fig. 8.

Program lines 2600 to 2699 make up a subroutine designed to obtain the X and Y coordinates of a given Point. This code corresponds to boxes 152, 154, 160, 162, 164, 166 and 168 of Fig. 8.

Program lines 3000 to 3339 constitute a Point and Stroke analysis and characterization routine embodying boxes 156, 158, 170, 172 and 173 of Fig. 8.

Program lines 3700 to 3790 make up a subroutine designed to compare the analyzed Strokes to a Stroke database. These program lines embody boxes 171 to 184 of Fig. 8.

Program lines 3810 to 3980 make up a subroutine which is designed to learn a new character. This code embodies boxes 186, 188, 190, 192 and 194 of Fig. 8.

Program lines 3060 to 3273 make up a subroutine designed for Stroke characterization purposes. This section of the code corresponds to Fig. 9.

Program lines 3060 to 3095 are used to calculate Point-to-Point slopes and embody box 220.

Program lines 3058, 3241 and 3262 are used to calculate a height above baseline (HABL) and correspond to box 222 of Fig. 9.

Programs lines 3253, 3270–3273 are used to normalize the Point height and slope and correspond to box 224 of Fig. 9.

Program line 3253 is used to store the height above baseline and embodies box 226 of Fig. 9.

The foregoing program can be stored in the memory of a microcomputer of microprocessor with a requirement of approximately 25K of machine memory, so that it can be seen that the use of the program does not use up a lot of expensive memory and is relatively fast in executing the program's operation. If the program is written in a language other than Basic, requiring less memory, such as assembly language, the size of the program can be made smaller.

Boxes 195 to 200 of Fig. 8 appear in logical places throughout the code.

A dictionary of the variables of the relevant code section is included as Appendix III.

With reference now to Fig. 13, a flowchart for the editing software ("Editor") demonstrated by Figs. 11A to 11I and described above is depicted. Once the Editor is loaded into the system (box 229), control of the screen is returned to the system. The system then proceeds in the normal manner described above to acquire Points and display them (box 230), convert the Points into Strokes (box 231), characterize each Stroke (box 232), and attempt to match the Stroke or Strokes with the database (box 233). In processing box 234, the system sends each Handwritten Symbol to the Editor to interpret and execute a command if necessary. At decision diamond 235, the Editor determines whether the Handwritten Symbol is an Editing Symbol or a Font Symbol. If the character is determined to be an Editing Symbol, the Editor proceeds to processing box 236 where it determines which Editing Symbol has been entered and executes the Editing Function. If the character is determined not to be an Editing Symbol, then the alphanumeric character corresponding to the handwritten entry is displayed at processing box 237. In an alternate configuration of the Editor, Font Symbols will only be accepted when the Editor is in the "Insert Mode". This structure insures that each Font Symbol is verified before being added to a document.

The Editor uses a variety of symbols designed to make editing on the system similar to, but much more efficient than, traditional editing with pencil and paper. These primary editing functions include, but are not limited to:

DELETE symbol—"______" A horizontal line drawn through a character or characters. The Editor will remove the underlying characters and reformat the text.

ADJUST MARGINS symbol—"|" A vertical line longer then the height of one line on the display. The Editor will adjust the margin to the indicated position and reformat the text.

INSERT symbol—"^" A caret drawn at the Point where text is to be added. The Editor displays an input writing line (Fig. 11B) and when input is recognized inserts it into the text.

MARK TEXT symbols—"<" and ">" A less than and greater than symbol drawn at the beginning and end of a block of text. The marked text is displayed in reverse video and then special block functions can be performed.

DELETE MARKED TEXT—A delete symbol drawn within marked text will erase the marked text and a reformat.

MOVE MARKED TEXT—An insert symbol drawn anywhere within the text moves the marked text to the indicated position, deletes it from its original position and reformats the text.

REPLACE MARKED TEXT—An insert symbol drawn within the marked text displays an input

line and replaces the marked text with the inputted text.

The Editing Symbols described above can be changed to the particular Editing Symbols preferred by each user, thereby customizing the Editor and preventing new users from having to learn unfamiliar Editing Symbols.

Further modifications and enhancements to the present system will be apparent to those skilled in the art. For example, the common characteristics of each Font Symbol could be extracted and organized into a synthetic symbol. The synthetic symbol's characteristics could then be exaggerated to maximize their variance from all other synthetic symbols. This would create a very compact, optimal database. On the other hand, as an example, a database created by the described preferred embodiment of the invention usually results in two to three different characterizations for each symbol.

The system has numerous useful applications, almost without limitation. The most obvious applications are text editing and filling out and modifying forms. Some of the many other applications that may not come to mind as readily are writing in languages utilizing large numbers of symbols like Japanese or Chinese, writing in Arabic and similar languages made up of a limited number of complex symbols; writing chemical equations, including those involving organic compounds; writing music (a "window" with five parallel lines can be provided for musical applications); writing symbols and codes for graphic manipulation of data, including the transfer of graphic data to a spreadsheet; in education, as where predetermined questions are presented on the screens and the answers written in long-hand; as in teaching mathematics, as when numbers are manually inserted in equations and the equation analyzed to determine the result using those numbers; in CAD/CAM applications involving symbols, geometric shapes and the like.

We have described a particular preferred embodiment of a keyboardless computer on which usual computer functions are performed by writing in a normal manner with a pen-like stylus on an input screen placed directly over a flat display. The keyboardless computer illustrated is ideally configured for use by non-keyboard oriented individuals, by keyboard individuals for whom the utility of the computer is enhanced, and in various settings and applications in which keyboard entry is awkward or impossible. Computer-based information and application software can be loaded into a portable device for later viewing, manipulation of text and data, and adding new text and data in a normal handwriting mode. Thereafter the user may transmit this computer text to another computer, a similar device, an external electronic storage device, a hard copy printer, or through a telecommunications system. The computer is capable of recognising Handwritten Symbols with a high degree of accuracy and of "learning" individual styles of handwriting. The data and commands can be input with the aid of a stylus.

Although the invention has been described in terms of a selected preferred embodiment encompassing the apparatus and methods aspects of a keyboardless computer system, the invention should not be deemed limited thereto, since other embodiments and modifications will readily occur to one skilled in the art.

APPENDIX I

```
'h' (1) ns = 1 x = (-17..0..19) y = (-6..18..60)
        1) np = 20 x = (-17..0..19) y = (-6..18..60)
              5,        42
            -57,        34
            -85,        26
            -71,        19
            -76,        10
            -64,         3
            -89,        -3
            -72,        -9
            -85,       -17
            -72,       -24
             47,       -17
             56,       -11
             43,        -3
             39,         3
             11,         7
             -9,         5
            -57,        -3
            -71,       -11
            -71,       -18

'H' (1) ns = 2 x = (-27..0..37) y = (-20..18..61)
        1) np = 14 x = (-27..-12..-2) y = (-20..22..61)
             -9,        37
              4,        38
            -39,        32
            -48,        25
            -58,        16
            -71,         8
            -72,         2
            -64,        -4
            -89,       -11
            -72,       -17
            -80,       -24
            -89,       -30
            -80,       -36
        2) np - 20 x = (-21..10..37) y = (-15..16..59)
             27,        43
            -80,        37
            -71,        29
            -98,        23
            -80,        16
            -78,         8
            -85,         1
            -89,        -6
            -80,       -12
            -80,       -18
            -64,       -25
            -64,       -31
             48,       -25
             56,       -18
             48,       -12
             64,        -6
             89,         1
           -124,         0
           -124,        -2
```

APPENDIX III

A1 - angle of slope between Point M and M1
A2 - previous angle
DA1 - change in angle (A1-A2)
SEND - flag for end of Stroke
ZX - double precision X position summation by arc length weighting
ZY - double precision Y position summation by arc length weighting
ZA - double precision angle summation by arc length weighting
ZDA - double precision change in angle summation
ZL - double precision length
NSP - number of Points in Stroke, counter
NSTRK - number of Strokes
XX - maximum X
XN - minimum X
YX - maximum Y
YN - minimum Y
M1 - Pointer to previous Point
M2 - (1) minimum number of Points for which to adjust for 2π angle
M3 - (2) minimum number of Points for which to calculate change in change in angle
NLONG - counter (not used)
M - Pointer to Point being considered
NP - number of Points
X - array of X coordinate
Y - array of Y coordinate
DX - change in X
DY - change in Y
ADX - absolute value of DX
ADY - absolute value of DY
DS - pseudo arc length ADX + ADY
AY - calibration multiplier for vertical (Y) direction, e.g. AY ' 0 mean coordinate system true downward
ADA - running average change in angle
DA2 - previous change in angle
ASTRT - array for starting angle of Stroke
SZL - single precision Stroke length
SNO - array of number of the 1st Point for each Stroke
LHT5 - height between lines divided by 5 in display coordinates used as a measure
SL - array of Stroke lengths sealed to height between lines
AEND - array of angles at end of Strokes
SC - array of angle changes for eachs troke
AZA - average angle through Stroke
SX - array of averages centroid X coordinate for each Stroke
SY - array of averages centroid Y coordinate for each Stroke
SA - array of averages angle (slope) coordinate for each Stroke
SXX - array of maximum X coordinate for each Stroke
SXN - array of minimum X coordinate for each Stroke
SYN - array of minimum Y coordinate for each Stroke
SYX - array of maximum Y coordinate for each Stroke
DBUG - debugging flag for printing
INPUT - string of input bytes to be rejected
XP, YP - input coordinates
XPA - average 2 Points in X direction
YPA - average 2 Points in Y direction
XPO - initial X average on pan down
YPO - initial Y average on pan down
I - display pixel corresponding to X coordinate input
J - display pixel corresponding to Y coordinate input
N - counter of input Points
MM - counter for timing pan up
NU - counter for timing pan up

CLAIMS

1. An input/output or I/O device for providing input signals to a utilization device that are representative of a graphical character produced by a user and for receiving output signals from the utilization device and providing a visual representation of the output signals, said I/O device comprising:

(a) a display means for providing a visual/display of a graphical character in response to output signals provided by the utilization device;

(b) an input screen means for producing a train of input signals as a graphical character or symbol is sketched by a user, said input screen means comprising a translucent input screen having top and bottom surfaces, said bottom surface being disposed with respect to said display surface such that the visual display produced by said display means is visible from said top surface of said input screen; and

(c) means for determining the identity of said graphical character and providing output signals representative thereof to said display means.

2. An I/O device as claimed in claim 1 wherein said display means comprises a display surface on which the graphical character is produced; and

said input screen means bottom surface is disposed in close proximity above said display surface such that the visual display produced by said display means is visible from the top side of said input screen.

3. An I/O device as claimed in claim 1 wherein said input screen means includes a substantially flat plate, and wherein said input screen top surface is the top surface of said plate.

4. An I/O device as claimed in claim 3 and further including an including an indicating means operated by a user for cooperating with said input screen top surface such that said input screen means generates said train of input signals when said indicating means is in a predetermined position with respect to said plate top surface.

5. An I/O device as claimed in claim 4 wherein said input screen means generates an input signal when said indicating means is in contact with said input screen top surface.

6. An I/O device as claimed in claim 5 wherein:

(a) said plate has electrical characteristics such that the voltage at any Point thereon varies in a known manner with the distance from the location at which a voltage is applied thereto;

(b) said input screen means further includes means for selectively sequentially applying a voltage along one Cartesian axis of said plate and then along the other Cartesian axis of said plate; and

c) said indicating means is elongate and has a tip at one end thereof and includes a voltage detecting means that provides a signal representative of the voltage present where said tip contacts said input screen plate.

7. An I/O device as claimed in claim 5 wherein:

(a) said plate has electrical characteristics such that the voltage at any Point thereon varies in a known manner with the distance from the location at which a voltage is applied thereto;

(b) said input screen means further includes means for selectively sequentially sampling at an origin thereon the voltage along one Cartesian axis of said plate and then along the other Cartesian axis of said plate; and

(c) said indicating means is elongate with a tip at one end thereof and include a means for applying a known voltage to said tip; said sampling means detecting the voltages at said origin when said tip contacts said input screen plate and provides a signal representative of the detected voltages.

8. An I/O device as claimed in claim 2 wherein said display screen has a display area on which graphic symbols can be generated and a key input area by which command signals for the utilization device can be generated when contact is made with the corresponding area thereabove on said input screen.

9. A handwriting character recognition apparatus comprising:

(a) a display means having a display surface for providing a visual display of graphical characters in response to a plurality of display signals;

(b) a writing surface means for receiving graphical characters handwritten by a user, said writing surface means being disposed in close proximity with respect to said display means and comprised of a substantially translucent surface which has positional coordinates associated therewith;

(c) an indicating means operated by a user in connection with said writing surface means for producing writing signals in cooperation with said writing surface means;

(d) position detecting means for converting said writing signals into locating signals representative of the positional coordinates of said indicating means with respect to said writing surface;

(e) a processing means for receiving and storing said locating signals as they are produced and for generating said display signals such that said display means can graphically display the graphical character after it has been produced by said indicating means; said processing means being comprised of a character recognition means for comparing said produced locating signals

representative of a graphical character with a plurality of stored database signals for identifying said graphical character.

10. A handwriting character recognition apparatus as claimed in claim 9 wherein said writing surface means includes a substantially flat plate which produces a signal when said indicating means is in contact therewith.

11. A handwriting character recognition apparatus as claimed in claim 9 and further including means for converting said positional coordinates into relational coordinates.

12. A handwriting character recognition apparatus as claimed in claim 11 and further including means for dividing up a stream of locating signals into sections representative of a Stroke; and means for calculating Stroke characteristics which can be compared with Stroke characteristics of the database signals.

13. A handwriting character recognition apparatus as claimed in claim 9 wherein said processing means further includes means for updating said stored database signals with said produced locating signals when no correct graphical character is identified.

14. A method of recognizing a Handwritten Symbol comprising the steps of:

(a) generating a stream of locating signals for Points by moving a stylus about locations proximate to a writing surface means that generates a signal representative of said location of said stylus;

(b) dividing up a stream of locating signals into sections representative of a Stroke;

(c) calulating Stroke characteristics;

(d) comprising said calculated Stroke characteristics with Stroke characteristics previously stored in a database; and

(e) determining the best comparison and indicating if said best comparison is good enough to be a match.

15. Apparatus for recognizing Handwritten Symbols and displaying Handwritten and Font Symbols on a screen comprising:

(a) A visual display screen having the graphic capability to display Font Symbols and execute predetermined commands;

(b) Hand-held means to write or draw Handwritten Symbols on or over said screen;

(c) Means to display a true representation of the Handwritten Symbols on said screen as they are created;

(d) Digitizing means to sense the position of said hand-held means and to convert the same into a series of electrical signals defining the position, size and shape of each Handwritten Symbol;

(e) Means to compare predetermined characteristics of each digitized Handwritten Symbol with a database of predetermined characteristics of Font Symbols to assess the identity of each Font Symbol;

(f) Means to convert the Handwritten Symbol to a predetermined Font Symbol or a command and to display the Font Symbol on the screen in close proximity to the screen area in which the Handwritten Symbol was originally entered or to execute the command, as the case may be.

16. Apparatus as set forth in Claim 15, including means to display predetermined fonts of symbols in the form of text or forms on said screen, whereby the Handwritten Symbols can be used to input information to complete a form or to edit a predetermined text.

17. Apparatus as set forth in Claim 15, wherein said apparatus comprises a portable unit having dimensions up to about 400mm by 400mm by 100mm, a weight of up to about 7 kilograms and is adapted to include a self-contained power supply.

18. Apparatus as set forth in Claim 15, including means to display predetermined symbols in a font and command-representing symbols on said screen, to permit handwritten symbols representing each predetermined symbol to be manually defined on the screen and means to identify the manually defined symbol with the corresponding predetermined symbol in said database.

19. Apparatus as set forth in Claim 18, including means to modify said database when a Handwritten Symbol causes an erroneous Font Symbol to be displayed on the screen or command to be performed, as the case may be.

20. Apparatus as set forth in Claim 16, including means to create a window of empty space on said screen upon a predetermined command for entering Handwritten Symbols in said window.

21. Apparatus as set forth in Claim 15, including areas on said screen sensitive to touch to cause predetermined functions to be performed.

22. Apparatus as set forth in Claim 15, wherein said screen is substantially flat and adapted to be used in a substantially horizontal position.

23. Apparatus as set forth in Claim 15, including Softkeys on said screen and means to execute operational commands in response to touching any of said Softkeys.

24. A microprocessor-based process for recognition, translation and display of Handwritten Symbols and execution of commands comprising the steps of:

(a) Creating a personalized database for each individual user of the process by having the user initially input a Handwritten Symbol for each character corresponding with a font to be displayed or command to be executed;

(b) Determining a unique set of characteristics to characterize each such symbol and storing it in the database;

(c) Writing or drawing Handwritten Symbols on a computer screen with a stylus;

(d) Digitizing each Handwritten Symbol to identify the x,y coordinates of a multiplicity of Points defining the Symbol;

(e) Processing the digitized characteristics of each Handwritten Symbol to determine the predetermined characteristics of the Symbol;

(f) Search the database to find a "hit," a perfect or the closest correspondence to the characteristics;

(g) Displaying the Font Symbol or performing the command most closely associated with the "hit" characteristics.

25. A process as set forth in Claim 24, including the additional steps of:

(a) Determining if the Font Symbol displayed or command performed is an error; and

(b) If there is an error, re-inputting a Handwritten Symbol for the desired Font Symbol or command to modify the database.

26. A process as set forth in Claim 24, including the additional step of:

(a) Substantially contemporaneously displaying the Handwritten Symbol on the screen as it is being written or drawn.

27. A process as set forth in Claim 24, wherein steps (b) and (c) involve a determination of the (i) length, (ii) average slope, (iii) centroid height above base line, (iv) curvature (rate of change in slope) and (v) comparison of the location of the centroid of each Stroke with the centroid of the Handwritten Symbol.

28. A process as set forth in Claim 27, wherein step (f) comprises:

(a) Determining the value of each of the five characteristics;

(b) Subtracting the database value of the characteristic from the newly-measured value;

(c) Scaling the absolute values of each measurement to make each of the five measurements reasonably equivalent to the others;

(d) Adding all five values;

(e) If a predetermined threshold is exceeded, discarding the comparison and creating an error message; and

(f) Searching the database to find a numerical "hit", or, if not feasible, the closest numerical match between the database value and the newly inputted value.

29. A process as set forth in Claim 24, wherein the error rate is 5% or less.

30. A process as set forth in Claim 28, wherein the program for step (f) requires less than 6K of machine memory.

31. A process as set forth in Claim 24, wherein said Handwritten Symbols include Editing Symbols and said process includes recognition of the Editing Symbols drawn on the screen and executing the Editing Functions represented.

32. A process as set forth in Claim 24, in which text in Font Symbol format is displayed on the screen, and including the steps of:

(a) Creating a window on the screen proximate to, but not overlying, an area of said text to be edited;

(b) Inputting and displaying said Handwritten Symbols in said window;

(c) Displaying the Font Symbols corresponding to said Handwritten Symbols in close proximity to said Handwritten Symbols.

33. A process as set forth in claim 24, including the steps of:

(a) Creating one or more Softkeys on said screen to cause the computer to perform operational functions; and

(b) Touching one or more of said Softkeys to execute the corresponding functions.

34. A process as set forth in Claim 24, including the steps of determining the x,y coordinates of from about 4 to 8 Points per mm and from about 100 to 200 points per second in characterizing said Handwritten Symbols.

Published 1988 at The Patent Office, State House, 66/71 High Holborn, London WC1R 4TP. Further copies may be obtained from The Patent Office, Sales Branch, St Mary Cray, Orpington, Kent BR5 3RD. Printed by Burgess & Son (Abingdon) Ltd. Con. 1/87.